UNIVERZA V LJUBLJANI

FAKULTETA ZA RAČUNALNIŠTVO IN INFORMATIKO

Boštjan Hren

Ocenjevanje zanesljivosti regresijskih napovedi na podatkovnih tokovih

DIPLOMSKO DELO

UNIVERZITETNI ŠTUDIJ
RAČUNALNIŠTVA IN INFORMATIKE

MENTOR: prof. dr. Igor Kononenko

Ljubljana, 2016

Fakulteta za računalništvo in informatiko izdaja naslednjo nalogo:

Tematika naloge:

Razvitih je že precej metod ocenjevanja zanesljivosti in se jih uporablja s klasičnimi klasifikacijskimi in regresijskimi modeli strojnega učenja. Podatkovni tokovi predpostavljajo kratke medprihodne čase novih primerov, zato morajo biti algoritmi za tvorjenje napovedi algoritmično hitri, preprosti. Pogost pristop je uporaba drsečih oken, kjer se hrani določena zgodovina prejetih primerov, nad katerimi se izvaja učenje.

V okviru diplomske naloge implementirajte in testirajte tvorbo empirične distribucije napake nad drsečim oknom z različnimi metodami, kot so npr. metoda maksimalnega verjetja, stremljenja in lokalnih okolic. Ocene o zanesljivosti posameznih novih napovedi prikažite v obliki napovednih intervalov. Razvite metode ovrednotite na več realnih podatkovnih množicah in z več različnimi algoritmi strojnega učenja tako, da uporabite pokrivno verjetnost, relativni povprečni interval in kombinirano statistiko. Naredite tudi časovno primerjavo učnih algoritmov z in brez uporabe ocen zanesljivosti (faktor upočasnitve) ter primerjajte dosežene točnosti pri enakih časovnih omejitvah (učinkovitost)

.

IZJAVA O AVTORSTVU DIPLOMSKEGA DELA

Spodaj podpisani Boštjan Hren sem avtor diplomskega dela z naslovom:

Ocenjevanje zanesljivosti regresijskih napovedi na podatkovnih tokovih

S svojim podpisom zagotavljam, da:

- sem diplomsko delo izdelal samostojno pod mentorstvom prof. dr. Igor Kononenka,
- so elektronska oblika diplomskega dela, naslov (slov., angl.), povzetek (slov., angl.) ter ključne besede (slov., angl.) identični s tiskano obliko diplomskega dela in
- soglašam z javno objavo elektronske oblike diplomskega dela v zbirki "Dela FRI".

V Ljubljani, dne Podpis avtorja:

Iskreno se zahvaljujem svojemu mentorju prof. dr. Igor Kononenku, da mi je omogočil opravljanje diplomske naloge in mi z nasveti in strokovno pomočjo pomagal pri delu. Zahvaljujem se tudi dr. Darko Pevcu za dodatno vsestransko pomoč pri izdelavi in snovanju diplomske naloge.

Nenazadnje se zahvaljujem tudi staršem in sestri, ki so me podpirali, spodbujali in niso izgubili upanja vame. Posebno zahvalo posvečam punci Mojci za vso podporo, razumevanje in pomoč v času pisanja diplomske naloge.

Kazalo

Seznam uporabljenih kratic

kratica	angleško	slovensko
FIFO	first in, first out	prvi noter, prvi ven
MLE	maximum likelihood estimation	metoda maksimalnega verjetja
kNN	k-nearest neighbours	k-najbližjih sosedov
LR	linear regression	linearna regresija
GLM	generalized linear model	posplošeni linearni model
RT	regression tree	regresijsko drevo
RF	random forest	naključni gozdovi
SVM	support vector machine	metoda podpornih vektorjev
ANN	artificial neural network	umetne nevronske mreže
RBFN	radial basis function network	mreža radialnih baznih funkcij
RMSE	root mean squared error	koren srednje kvadratne napake

Povzetek

Z današnjo tehnologijo je mogoče preprosto neprekinjeno zbiranje podatkov. Kljub temu predstavlja pridobivanje znanja iz potencialno neskončnih podatkovnih tokov odprt problem. Zaradi določenih omejitev morajo biti metode za procesiranje podatkov dobro zasnovane, prostorsko učinkovite, računsko enostavne in hitre. Pogosto se analiza opravi na fiksni zgodovini podatkovnega toka, ki je določena z drsečim oknom. Kvaliteto napovedi algoritmov običajno ocenimo glede na njihovo povprečno točnost. Vendar, ko imamo opravka s podatki v realnem času, je lahko prav tako pomembna zanesljivost izhodnih vrednosti.

V diplomskem delu obravnavamo ocenjevanje zanesljivosti posameznih napovedi pri učenju na podatkovnih tokovih. Raziskali smo različne metode, ki tvorijo intervalne ocene zanesljivosti s pristopom maksimalnega verjetja, stremljenja in lokalnih okolic, za delo na neprekinjenih dinamičnih podatkih.

Metode smo implementirali z različnimi algoritmi strojnega učenja in jih preizkusili na več realnih in umetnih regresijskih problemih pri različnih velikostih drsečega okna. Uspešnost intervalnih cenilk smo ovrednotili z ocenami pokrivna verjetnost, relativni povprečni napovedni interval in kombinirana statistika. Primerjali smo izvajalne čase učnih algoritmov z in brez ocen zanesljivosti ter dosežene točnosti napovedi pri enakih časovnih omejitvah. Rezultate analiziramo tudi vizualno.

Ključne besede: strojno učenje, ocenjevanje zanesljivosti, napovedni intervali, podatkovni tokovi, regresija, napovedovanje

Abstract

Title: Reliability estimation of regressional predictions on data streams

With today's technology it is easy to collect data continuously. Still, how to extract knowledge from potentially infinite data streams remains an open problem. Because of specific constraints, stream processing methods have to be well designed, space-efficient, computationally simple and fast. Typically, data analysis is done on a fixed history of the data stream defined by a sliding window. We usually define the quality of predictions by their average accuracy. However, when dealing with real-time data it can be also important to know the reliability of the models' output values.

In this thesis we deal with online reliability estimation of individual predictions on data streams. We consider different interval reliability estimators based on maximum likelihood, bootstrap and local neighborhood approach for working on continuous dynamic data.

We implement these methods on different regression models and test them on several real and artificial regression problems with various sizes of the sliding window. Performance of the interval estimates are evaluated using the estimates of prediction interval coverage probability, the relative mean prediction interval and the combined statistic. We compare the execution times of learning algorithms with and without the reliability estimates as well as their prediction accuracy when given the same time constraint. We also analyze results visually.

Keywords: machine learning, reliability estimation, prediction intervals, data stream, regression, predicting

Poglavje 1 Uvod

V obliki podatkovnih tokov se dnevno generira ogromna količina podatkov. Viri so lahko komunikacijska omrežja, internetni promet, spletne transakcije, nadzorni sistemi, industrijski procesi ali druga dinamična okolja. Samodejno pridobivanje znanja je zahtevna naloga zaradi številnih omejitev pri delu s podatkovnimi tokovi. Posledično napovedni modeli podajo manj zanesljiv odgovor.

Da bi odkrili zakonitosti ali vzorce iz podatkovnih tokov, so potrebne dobro zasnovane metode učenja. Mnogo učnih algoritmov strojnega učenja je mogoče prilagoditi za delo z ne-statičnimi podatki. Ko imamo opravka s potencialno neskončnim tokom podatkov, je pogost pristop uporaba drsečih oken, kjer upoštevamo le omejeno število preteklih primerov za učenje novih modelov. Naprednejši inkrementalni algoritmi lahko upoštevajo celotno zgodovino in kljub temu ostajajo prostorsko in časovno učinkoviti.

Navadno nas zanima točnost njihovih napovedi, vendar je njihova zanesljivost lahko enako ali celo bolj pomembna. Razvite so številne metode za ocenjevanje zanesljivosti posameznih napovedi. V diplomski nalogi smo se osredotočili na intervalne ocene zanesljivosti, ki podajo napoved v obliki napovednih intervalov. Negotovost v napovedi modelov je lahko posledica različnih omejitev dela s podatkovnimi tokovi, ki jih opisujemo v nadaljevanju.

1.1 Domena učenja

Običajna paketna obdelava podatkov [22] je zelo učinkovita pri obdelavi podatkov velikega obsega, kjer imajo sistemi dostop do celotnega nabora podatkov in čas procesiranja ni pomemben. Nasprotno pa obdelava podatkovnih tokov vključuje stalen, potencialno neomejen dotok podatkov. V mnogih primerih je količina podatkov prevelika za shranjevanje v pomnilniku in z večanjem volumna je te nemogoče prebrati več kot enkrat. Ko je primer obdelan, je zavržen ali arhiviran - shranjen v pomnilniku, ki je dosti manjši od velikosti vhoda.

Vhodni podatki prihajajo zelo hitro in z različnimi stopnjami posodabljanja. Učni algoritem nima vpliva na vrstni red primerov, ki jih vidi, zato mora model posodobiti sproti, ko obdela posamezen primer. Dodatna zaželena lastnost je, da lahko model poda napoved v vsakem

trenutku, tudi v fazi učenja. Da se podatki lahko obdelajo in analizirajo v skoraj realnem času, so izračuni preprosti. Zaradi hitrih odločitev so odgovori približni.

Pogosto se podatki spreminjajo skozi čas na nepredvidljive načine, kar lahko privede do spremembe ciljnega koncepta [15] in [4]. To predstavlja težavo, saj napovedi s časom postanejo manj točne. Modele za napovedovanje na podatkovnih tokovih je mogoče zgraditi in posodobiti z uporabo različnih pristopov. *Periodični pristop* pomeni model ponovno naučiti po določenem času, *inkrementalni pristop* pa posodobiti model vsakič ob prihodu novega podatka. Tretji, *reaktivni pristop*, pomeni spremljati spremembe in model zgraditi ponovno, ko ta več ne ustreza podatkom.

Algoritme za obdelavo podatkovnih tokov lahko ovrednotimo glede na hitrost izvajanja, porabo prostora in točnost njihovih napovedi. Za nas je poleg uspešnosti ocen zanesljivosti pomembna predvsem časovna zahtevnost metod.

1.2 Predznanje

V tem razdelku na kratko opišemo nekatere pristope in tehnike, ki jih uporabljajo obravnavane metode za ocenjevanje zanesljivosti.

Pristop *drsečih oken* temelji na predpostavki, da je koristneje uporabiti novejše kot starejše podatke [4]. Obdelava podatkov se izvaja le na fiksni zgodovini podatkovnega toka. Najpreprostejši način je uporaba okna fiksne širine, pri katerem upoštevamo le zadnjih w najnovejših primerov shranjenih v pomnilniku. V tem pogledu se okno obnaša podobno kot podatkovna struktura prvi noter, prvi ven (FIFO). Drseča okna so tudi tehnika pozabljanja starih informacij.

Taylorjeva razširitev (angl. *Taylor expansion*) je metoda aproksimacije neskončno odvedljive funkcije okoli dane točke x_0 kot neskončna vsota njenih višjih odvodov [2]. V majhni okolici točke x_0, lahko na funkcijo $f(x)$ gledamo kot na linearno funkcijo, zato pogosto opustimo člene višjega reda. Tako dobimo linearno Taylorjevo aproksimacijo prvega reda.

Metoda maksimalnega verjetja (angl. *maximum likelihood estimation*) je metoda za ocenjevanje vrednosti enega ali več parametrov statističnega modela [25]. Namen metode MLE je najti najbolj verjetne vrednosti porazdeljenih parametrov, ki najbolje ustrezajo opazovanim podatkom. To naredi z maksimiziranjem vrednosti t.i. funkcije verjetja.

Stremljenje (angl. *bootstrapping*) je statistični pristop, ki temelju na naključnem vzorčenju z vračanjem [12]. Omogoča razmnoževanje učnih primerov s ponovnim vzorčenjem primerov, ki smo jih že videli. Tako generiramo nove množice, na katerih lahko izračunamo želeno statistiko (npr. varianco) in ocenimo njeno porazdelitev.

1.3 Sorodne raziskave

Z analizo zanesljivosti posameznih napovedi na podatkovnih tokovih se ukvarjajo številne raziskave. V delu [32] so predstavljeni različni pristopi k podajanju točkovnih ocen zanesljivosti regresijskih napovedi na podatkovnih tokovih. S praktično uporabo teh cenilk na sistemu za napovedovanje porabe električne energije se ukvarjajo v [8]. Avtorji dela [7] obravnavajo tudi nekatere metode za intervalno ocenjevanje zanesljivosti in jih uporabijo v kombinaciji z najsodobnejšimi pristopi za prepoznavanje sprememb ciljnega koncepta.

1.4 Pregled vsebine

Diplomsko delo obsega šest poglavij. V naslednjem poglavju na kratko predstavimo različne regresijske napovedne modele strojnega učenja. Omenimo tudi prilagoditve učnih algoritmov za inkrementalno delo na podatkovnih tokovih. Tretje poglavje se osredotoča na metode za intervalno ocenjevanje zanesljivosti z napovednimi intervali. Na kratko jih predstavimo in analiziramo njihovo časovno zahtevnost. Opis problema in uporabljenih statistik za vrednotenje intervalnih ocen zanesljivosti je podan v četrtem poglavju. V petem poglavju podamo rezultate testiranj in njihovo analizo. V zadnjem, šestem poglavju, komentiramo rezultate ter podamo naše zaključke in napotke za nadaljnje delo.

Poglavje 2 Napovedno modeliranje

Napovedno modeliranje poskuša najti matematično razmerje med odvisno ciljno spremenljivko in eno ali več neodvisnimi spremenljivkami (atributi). Cilj je analizirati trenutne podatke in zgraditi model za napovedovanje prihodnjih oz. neznanih izidov.

V splošnem lahko naloge strojnega učenja razdelimo v dve glavni kategoriji, odvisno od želene izhodne vrednosti naučenega sistema. Kadar ima ciljna spremenljivka diskretne vrednosti govorimo o klasifikaciji. Podatki so razdeljeni v dva ali več razredov in naloga klasifikatorja je, da določi eno (ali več) oznak razreda za vsak nov vhod. Pri regresiji je izhodna vrednost zvezna. V splošnem je lahko vrednost odvisne spremenljivke katerakoli realna vrednost. Cilj napovednega modela je, da napove neznano vrednost ciljne spremenljivke kot funkcijo atributov vhodnih podatkov. V tej diplomski nalogi smo se osredotočili na regresijske napovedne modele.

2.1 Regresijsko napovedovanje

Pri regresijskem napovedovanju izhajamo iz množice opazovanj $(\overrightarrow{x_i}, y_i), i = 1..n$, kjer so $\overrightarrow{x_i}$ vektorji diskretnih ali zveznih vhodnih vrednosti (atributov) in so y_i zabeležene vrednosti ciljnih spremenljivk. Naloga učnega algoritma, da iz učnih primerov čim bolje oceni parametre (ne)linearne regresijske funkcije, ki jo uporabi za napovedovanje novih primerov.

V nadaljevanju so opisani najbolj razširjeni regresijski učni algoritmi strojnega učenja, ki jih je mogoče prilagoditi za delo na podatkovnih tokovih. Zadnja dva predstavljena modela strojnega učenja, mreža radialnih baznih funkcij in bagging, ne bomo preizkusili kot samostojne modele. Omenimo jih, ker jih uporabljajo nekatere obravnavane metode za ocenjevanje zanesljivosti.

2.1.1 K-najbližjih sosedov (kNN)

Algoritem k-najbližjih sosedov (angl. *k-nearest neighbors*) je primer lene metode učenja. Učenje predstavlja le shranjevanje vseh učnih primerov v pomnilniku. Ko prispe nov primer, algoritem poišče k učnih primerov, ki so mu najbolj podobni (najbližji). V regresiji je običajno napoved povprečje vrednosti odvisnih spremenljivk k najbližjih sosedov. Alternativni pristop je izračun uteženega povprečja glede na oddaljenost vsakega od k sosedov. Za zvezne

spremenljivke je najpogosteje uporabljena mera Evklidska razdalja (angl. *Euclidean distance*) v atributnem prostoru.

Najbolj preprost inkrementalni algoritem kNN, le shrani vsak nov primer. Z omejenim časom in pomnilnikom to ni mogoče. V večini primerih velja, da so novejši podatki bolj uporabni kot starejši, zato lahko uporabimo pristop drsečega okna, kjer ohranimo le zadnjih w primerov. Za uporabo v posebnih aplikacijah so bile razvite bolj zapletene inkrementalne metode, ki so predstavljene v [13] in [19].

2.1.2 Linearna regresija (LR)

Linearna regresija (angl. *linear regression*) poskuša določiti vrednost odvisne spremenljivke y kot linearno kombinacijo vrednosti neodvisnih spremenljivk x_i (vrednosti atributov):

$$\hat{y}(\vec{x}) = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \cdots + \beta_m x_m \tag{2.1}$$

Naloga algoritma se je naučiti vrednosti regresijskih parametrov β (imenovani tudi uteži) tako, da je napovedana vrednost čim bližje pravi vrednosti odvisne spremenljivke. Linearni regresijski modeli najpogosteje za ocenjevanje vrednosti parametrov uporabljajo metodo najmanjših kvadratov (angl. *least squares*), ki poskuša minimizirati razlike med opazovanimi (pravimi) vrednostmi y in napovedanimi vrednostmi $\hat{y}$.

Nasprotno od klasičnega pristopa, kjer poskušamo zmanjšati napako na celotni učni množici podatkov, različice linearne regresije za sprotno učenje, obravnavajo napako na vsakem primeru posebej. Za delo na neprekinjenih podatkovnih tokovih so bili razviti učinkoviti algoritmi linearne regresije, ki lahko v realnem času posodobijo parametre regresijske funkcije [26] in [37].

2.1.3 Posplošeni linearni model (GLM)

Posplošeni linearni model (angl. *generalized linear model*) predstavlja posplošitev linearne regresije. V najenostavnejši obliki linearni model predpostavlja linearno odvisnost med normalno porazdeljeno ciljno spremenljivko y in več neodvisnimi slučajnimi spremenljivkami x_i. Posplošeni linearni model dovoljuje, da ima odvisna spremenljivka poljubno porazdelitev in njen odziv na spremembe neodvisnih spremenljivk v osnovi ni linearen.

Ciljna spremenljivka je z vhodnimi spremenljivkami povezana preko posebne povezovalne funkcije (angl. *link function*) g:

$$\hat{y}(\vec{x}) = g(\beta_0 + \beta_1 x_1 + \beta_2 x_2 + \cdots + \beta_m x_m) \tag{2.2}$$

Ta podaja povezavo med linearno kombinacijo spremenljivk x_i in pričakovano vrednostjo ciljne spremenljivke.

Vrednosti parametrov so pridobljene z metodo maksimalnega verjetja (iz razdelka 1.2). S pristopi, ki omogočajo sprotne posodobitve uteži, lahko posplošeni linearni model uporabljamo inkrementalno [11].

2.1.4 Regresijska drevesa (RT)

Regresijska drevesa (angl. *regression trees*) so vrsta odločitvenih dreves, kjer lahko ciljna spremenljivka zavzame zvezne ali urejene diskretne vrednosti. Odločitveno drevo rekurzivno deli množico primerov tako, da podobne primere grupira skupaj. V tej drevesni strukturi vsako notranje vozlišče ustreza enemu od vhodnih atributov, medtem ko povezave predstavljajo podmnožice njihovih vrednosti. Vsak list v regresijskih drevesih predpisuje regresijsko enačbo (najpogosteje konstanto ali linearno funkcijo), ki iz vhodnih vrednosti primera napove izhodno vrednost ciljne spremenljivke.

Vsaka pod od korena do lista ustreza konjunkciji pogojev vrednosti atributov. Odločitvena drevesa se običajno gradijo od zgoraj navzdol, tako da se na vsakem koraku izbere atribut, ki po izbranem kriteriju najbolje razdeli množico primerov. Različni učni algoritmi uporabljajo različne mere za ugotavljanje najboljše delitve.

Večina algoritmov odločitvenih dreves za učenje potrebuje celotno množico podatkov. Najbolj preprost inkrementalni pristop je izgradnja novega drevesa, ko za vsak nov primer izvemo pravo vrednost odvisne spremenljivke [34]. Učinkovitejše metode omogočajo postopno posodabljanje odločitvenih dreves tako, da uporabijo le nove individualne primere za prestrukturiranje dreves [39]. Učni algoritmi za sprotno učenje odločitvenih dreves so bili prirejeni tudi za podatkovne tokove, ki se spreminjajo skozi čas [20] in [16].

2.1.5 Naključni gozdovi (RF)

Naključni gozdovi (angl. *random forests*) so primer učenja ansamblov (angl. *ensemble learning*), ki združuje napovedi več modelov odločitvenih dreves. Za izgradnjo posameznih dreves se uporablja prilagojen učni algoritem, ki omeji izbiro delitvenih vrednosti v vsakem koraku na naključno podmnožico atributov. Vsako drevo za nov primer poda napoved ciljne

spremenljivke. Pri regresiji je izhod modela naključni gozdovi povprečna vrednosti posameznih napovedi. Metoda učinkovito zmanjša varianco enega drevesa in je odporna na preveliko prileganje (angl. *overfitting*).

Za delo na podatkovnih tokovih so primerni naključni gozdovi s sprotnim učenjem [33]. Ti združujejo prirejeno metodo stremljenja in algoritme za sprotno učenje odločitvenih dreves z naključnim izborom atributov.

2.1.6 Metoda podpornih vektorjev (SVM)

Osnovna ideja klasifikacijske metode podpornih vektorjev (angl. *support vector machines*) je najti optimalno hiperravnino, ki ločuje vse primere enega razreda od primerov, ki pripadajo drugemu razredu. Optimalna hiperravnina maksimira odmik - oddaljenost ravnine od najbližjih primerov enega in drugega razreda. Primeri na robu, pravimo jih podporni vektorji (angl. *support vectors*), natančno določajo odločitveno funkcijo. Če množica primerov ni linearno ločljiva, se primeri preslikajo v višjo dimenzijo prostora, kjer je razrede mogoče jasno ločiti. Ta pristop imenujemo trik jedra (angl. *kernel trick*).

Razširitev metode podpornih vektorjev za regresijske probleme z uporabo nelinearne preslikave preslika primere v višje-dimenzijski prostor, kjer je problem mogoče rešiti z linearno regresijo. Cilj je najti odločitveno funkcijo, ki za največ ε (dovoljena meja odstopanja) odstopa od ciljnih vrednosti in je pri tem čim bolj ploska. Nasprotno kot pri klasifikaciji, poskuša algoritem rob minimizirati in hkrati čim bolj zmanjšati napako na učnih primerih.

Nadgradnje metode SVM, ki omogočajo sprotno učenje, lahko delujejo na zaporedju primerov in generirajo inkrementalne rešitve [10]. Za delo na velikih ali ne-statičnih množicah so bile razvite različne implementacije [21].

2.1.7 Umetne nevronske mreže (ANN)

Umetne nevronske mreže (angl. *artificial neural networks*) so statistični model učenja, ki delujejo po vzoru bioloških nevronskim mrež (možganov). Osnovni gradniki umetne nevronske mreže so umetni nevroni, ki predstavljajo enostavne matematične modele. Ti so organizirani v več nivojev: vhodni nivo (sprejme vhod), en ali več skritih nivojev ter izhodni nivo z enim samim nevronom, ki predstavlja ciljno spremenljivko.

Nevroni so med seboj povezani z numerično uteženimi povezavami. Izhod vsakega nevrona pri danih uteženih vhodih definira aktivacijska funkcija. Dodaten parameter prag določa nivo signala, pri katerem se nevron sproži. Naloga učnega algoritma je, da nastavi proste parametre

(uteži in pragove), tako da minimizira izbrano funkcijo napake. Za posodobitev uteži je najpogosteje uporabljen algoritem vzvratnega razširjanja napake (angl. *backpropagation of error*).

S pravo implementacijo umetne nevronske mreže omogočajo sprotno učenje na velikih množicah podatkov. Inkrementalne arhitekture nevronskih mrež so bile predlagane za različne aplikacije [9] in [30]. Te metode omogočajo, da model obravnava le nove podatke, hkrati pa upošteva že pridobljeno znanje.

2.1.8 Mreža radialnih baznih funkcij (RBFN)

Mreže radialnih baznih funkcij (angl. *radial basis function network*) so vrsta umetnih nevronskih mrež, ki za aktivacijske funkcije uporabljajo radialne bazne funkcije. Navadno si umetne nevronske mreže predstavljamo kot večnivojski perceptron (angl. *multilayer perceptron*), kjer vsak nevron prejme utežene vhode in se sproži, če je njihova vsota dovolj velika. En sam nevron perceptrona predstavlja preprost linearen model, vendar združeni v mrežo lahko rešijo zapletene nelinearne probleme.

Mreža z radialnimi baznimi aktivacijskimi funkcijami izračuna izhod s primerjanjem podobnosti med vhodnim primerom in primeri iz učne množice. Vsak RBFN nevron vsebuje prototip, ki je eden od učnih primerov. Kot mera podobnosti se najpogosteje uporabi evklidska razdalja. Vrednost ciljne spremenljivke prestavlja linearna kombinacija vrednosti baznih funkcij pri izračunani razdalji med novim primerom in prototipom. V osnovni strukturi mrežo sestavljajo trije nivoji.

Prednost mrež radialnih baznih funkcij v primerjavi z večnivojskim perceptronom je njihovo hitro učenje. Kadar obravnavamo obsežne zaporedne podatke lahko gradnjo modela še pospešimo z uporabo inkrementalnih učnih algoritmov [1].

2.1.9 Bagging

Metoda bagging (angl. tudi *boostrap aggregating*) s postopkom stremljenja generira več novih učnih množic z vzorčenjem primerov originalne učne množice. Vsaka nova množica je enake velikosti kot originalna množica, vendar ne vsebuje vseh primerov. Pri postopku vzorčenja se nekateri primeri lahko ponovijo večkrat, drugi pa sploh niso izbrani. V povprečju vsaka množica vsebuje okoli dve tretjini različnih primerov.

S pomočjo danega učnega algoritma se na vsaki množici nauči nov napovedni model. Skupna napoved $\hat{y}_{bag}(\vec{x})$ modela bagging je povprečje posameznih napovedi b notranjih modelov:

$$\hat{y}_{bag}(\vec{x}) = \frac{1}{b}\sum_{i=1}^{b} \hat{y}_i(\vec{x}) \tag{2.3}$$

Metoda je časovno zelo zahtevna. Kadar se učna množica s časom veča, lahko uporabimo sprotno različico algoritma bagging [27], ki nove primere v vzorčene učne množice vključuje sproti.

Poglavje 3 Ocenjevanje zanesljivosti napovedi

Za ovrednotenje regresijskih učnih algoritmov se običajno uporabljajo povprečne ocene uspešnosti, kot sta klasifikacijska točnost ali srednja kvadratna napaka. Kljub temu nam lahko dodatne informacije pomagajo bolje razumeti rezultate modelov. Mere za ocenjevanje zanesljivosti nam podajo zaupanje v pravilnost njihovih napovedi.

V splošnem ločimo med točkovnimi in intervalnimi ocenami zanesljivosti. Točkovne ocene lahko uporabimo s klasifikacijskimi in regresijskimi modeli strojnega učenja, medtem ko so intervalne ocene definirane le za regresijske probleme. Slednje ponujajo veliko boljši vpogled v eksperimentalne rezultate, zato smo se v diplomski nalogi posvetili intervalnemu ocenjevanju zanesljivosti.

3.1 Intervalno ocenjevanje zanesljivosti

Pri regresijskem problemu obravnavamo zvezno funkcijo f, ki slika iz atributnega prostora v realno vrednost izhodne spremenljivke y. Vrednost funkcije je lahko pokvarjena z dodatnim šumom ε:

$$y = f(X) + \varepsilon(X) \tag{3.1}$$

Kadar je varianca napake konstantna, govorimo o *homoskedastičnosti* (angl. *homoscedasticity*) podatkov. Večina regresijskih modelov temelji na tej predpostavki. Drugi pojem, ki opisuje razpršenost šuma, je *heteroskedastičnost* (angl. *heteroscedasticity*), ki pomeni, da se varianca slučajne napake spreminja glede na vrednost neodvisnih spremenljivk. V tem primeru je lahko poleg šuma podatkov vir napake tudi v strukturi samega modela.

Na točnost napovedi modela vplivajo dejavniki, kot so distribucija učnih primerov, pristranskost, občutljivost učnega algoritma na šum in drugo. Napako ocenimo z *varianco negotovosti modela* σ_m^2. Ena od mer negotovosti modela je *interval zaupanja* (angl. *confidence interval*), ki se ukvarja z natančnostjo ocene prave, vendar neznane, regresijske funkcije in zajema razpon verjetnih vrednosti za neznani parameter.

Za šum v podatkih so lahko različni vzroki: napake pri meritvah, težave pri branju podatkov, človeške napake, napačno klasificirani primeri... Te nepravilnosti v podatkih opišemo z *varianco šuma podatkov* σ_p^2. Varianca negotovosti modela in varianca šuma podatkov skupaj predstavljata *skupno varianco napovedi* σ_ε^2. Če predpostavljamo neodvisnost obeh komponent, velja:

$$\sigma_\varepsilon^2 = \sigma_m^2 + \sigma_p^2 \tag{3.2}$$

S skupno varianco napovedi je povezan pojem *napovedni interval* (angl. *prediction interval*). Ta je podan kot interval vrednosti, na katerem naj bi se z določeno verjetnostjo nahajala prava vrednost odvisne spremenljivke novega primera. Napovedni interval je širši in hkrati zajema interval zaupanja.

Najenostavnejši so konstantni napovedni intervali, ki jih dobimo direktno iz porazdelitve opazovanih vrednosti ciljne spremenljivke. Pri izbrani stopnji tveganja α interval vsebuje vrednosti med $(100 \cdot \alpha/2)$. in $(100 - 100 \cdot \alpha/2)$. percentilom odvisne spremenljivke, kar naj bi predstavljalo $(1 - \alpha) \cdot 100$ odstotkov vseh. Očitno je, da so takšni intervali neuporabni za dobro oceno zanesljivosti.

V nadaljevanju smo predstavili metode, ki tvorijo intervalne ocene zanesljivost v obliki napovednih intervalov z uporabo različnih pristopov. Oceno časovnih zahtevnosti metod zapišemo z O-notacijo, kjer je n število učnih primerov in m število atributov. Za model M z $O(M_L)$ označujemo časovno zahtevnost faze učenja in z $O(M_P)$ časovno zahtevnost napovedovanja.

3.1.1 Analitični pristop

Kadar imamo veliko učnih primerov, lahko naučimo model, ki nudi dobro aproksimacijo prave regresijske funkcije. Če predpostavimo tudi homoskedastičnost podatkov (šum je normalno porazdeljen s srednjo vrednostjo 0 in ima konstantno varianco), lahko intervalno oceno zanesljivosti posamezne napovedi ocenimo s preprostim analitičnim pristopom.

V primeru preproste linearne regresijske funkcije ene spremenljivke je napovedni interval podan kot:

$$\hat{y}(x) \pm t_{\alpha/2,n-2}\, \mathrm{S}_y \sqrt{1 + \frac{1}{n} + \frac{(x - \bar{x})^2}{\sum_{i=1}^{n}(x_i - \bar{x})^2}} \tag{3.3}$$

kjer je $t_{\alpha/2,n-2}$ kritična vrednost *Studentove t-porazdelitve* (angl. *Student's t-distribution*) pri gostoti verjetnosti $1 - \alpha/2$ in stopnji svobode $n - 2$. V enačbi vrednost S_y predstavlja standardi odklon , ki ga izračunamo kot:

$$\mathrm{S}_y = \sqrt{\frac{\sum_{i=1}^{n}(y_i - \bar{y}_i)^2}{n - 2}} \tag{3.4}$$

Na podoben način izračunamo napovedne intervale za večkratno linearno regresijo, vendar so zaradi večjega števila spremenljivk enačbe veliko bolj zapletene.

Če obravnavamo nelinearen problem, regresijsko funkcijo najprej lineariziramo. Uporabimo Taylorjevo razširitev prvega reda (iz razdelka 1.2), da dobimo linearno aproksimacijo funkcije, s pomočjo katere izračunamo napovedne intervale. O intervalih te vrste lahko beremo v večini literature o regresijskem napovedovanju, npr. [35].

Analitične napovedne intervale je mogoče izračunati za mnogo modelov, vendar ne za vse. Model najbližji sosedi in regresijska drevesa interpretirajo izhod drugače. Prav tako takšne teoretične formule niso na voljo za nekatere kompleksnejše nelinearne modele. Metodo z analitičnim pristopom zato tukaj le omenimo in se v nadaljevanju posvetimo pristopom, ki so splošni in jih lahko uporabimo z vsakim regresijskim napovednim modelom.

3.1.2 Maksimalno verjetje (ML)

V nasprotju z analitičnem pristopom tukaj ne predpostavimo homoskedastičnosti podatkov. Metoda maksimalnega verjetja (iz razdelka 1.2) poskuša varianco šuma oceniti kot funkcijo vhoda $\vec{x}$. Enako kot pri prvi metodi predpostavi, da je naučen model dober približek prave regresijske funkcije.

Varianco posamezne napovedi ocenimo z novim modelom, ki smo ga naučili s ciljnimi vrednostmi kvadratov residualov (razlike napovedi in pravih vrednosti odvisne spremenljivke): $(\hat{y}(\vec{x}) - y(\vec{x}))^2$ in je napovedni interval podan kot:

$$\hat{y}(\vec{x}) \pm z_{\alpha/2}\, \hat{\sigma}_{\varepsilon}^2(\vec{x}) \tag{3.5}$$

kjer $\hat{y}(\vec{x})$ predstavlja napoved osnovnega modela, z (z-vrednost) zgornjo $100 \cdot (1 - \alpha/2)$ odstotno točko standardne normalne porazdelitve s povprečjem 0 in standardnim odklonom 1 ter je $\hat{\sigma}_{\varepsilon}^2(\vec{x})$ napovedana varianca napovedi novega modela.

Časovna zahtevnost metode ML je odvisna od uporabljenega modela za napovedovanje variance napovedi. Najpogosteje se uporabljajo umetne nevronske mreže, ki so oblika ocenjevanja maksimalnega verjetja. Mi smo v ta namen uporabili časovno manj zahtevno mrežo radialnih baznih funkcij (iz razdelka 2.1.8). Za n residualov potrebuje model RBFN za učenje čas največ reda $O(n^2)$ in $O(M_P + n + 1)$ za napoved.

3.1.3 Stremljenje in maksimalno verjetje

Ko imamo malo podatkov ali opravka s težjim problemom ter šum ni preprost, je možen pristop, da vsako od komponent skupne variance ocenimo posebej. Razvitih je več metod za ocenjevanje negotovosti modela. Pogosto uporabljen je pristop stremljenja (iz razdelka 1.2), ki nudi dobre rezultate tudi na manjših množicah podatkov. Varianco šuma podatkov je moč oceniti z že omenjeno metodo maksimalnega verjetja. Kombinacijo obeh pristopov uporabljata metodi, ki ju povzemamo v nadaljevanju.

3.1.3.1 Heskes (BML-A in BML-A*)

V delu [18] predstavljena metoda negotovost modela oceni z varianco porazdelitve notranjih napovedi modela bagging (iz razdelka 2.1.9) za vsak primer posebej. Ob predpostavki, da so te normalno porazdeljene, uporabimo enačbo:

$$\hat{\sigma}_m^2(\vec{x}) = \frac{1}{b-1} \sum_{i=1}^{b} (\hat{y}_i(\vec{x}) - \hat{y}_{bag}(\vec{x}))^2 \tag{3.6}$$

kjer je b (tipično $b = 30$) število notranjih modelov, $\hat{y}_i(\vec{x})$ napoved i-tega modela ter $\hat{y}_{bag}(\vec{x})$ skupna napoved modela bagging kot povprečje vseh posameznih napovedi.

O varianci šuma podatkov sklepamo iz množice residualov. Za nepristransko oceno je bolje uporabiti residuale ločenih testnih primerov, ki niso bili uporabljeni za učenje modelov. Pri postopku vzorčenja s ponavljanjem vsi primeri hkrati niso vsebovani v vseh učnih množicah,

kar ponuja drugačno rešitev. Pri računanju razlik za posamezen primer se uporabi prirejena skupna napoved $\hat{y}_{bag(\vec{x})}(\vec{x})$:

$$\hat{y}_{bag(\vec{x})} = \frac{\sum_{i=1}^{b} q_i(\vec{x})\hat{y}_i(\vec{x})}{\sum_{i=1}^{b} q_i(\vec{x})} \tag{3.7}$$

kjer se upoštevajo le napovedi modelov, ki pri učenju niso obravnavali izbran primer: $q_i(\vec{x}) = 1$, če primer $\vec{x}$ ni bil vsebovan v učni množici modela M_i.

Pri predpostavki Gaussove porazdelitve šuma, varianco šuma podatkov ocenimo z metodo maksimalnega verjetja, pri čemer so ciljne vrednosti kvadrati residualov zmanjšani za ocenjeno varianco negotovosti modela za dan učni primer in navzdol omejeni z 0:

$$r^2(\vec{x}) \equiv \max(\left[y(\vec{x}) - \hat{y}_{bag(\vec{x})}(\vec{x})\right]^2 - \hat{\sigma}_m^2(\vec{x}), 0) \tag{3.8}$$

Tako naučen model nam napove oceno variance šuma podatkov $\hat{\sigma}_p^2(\vec{x})$. Ker je predpostavljena neodvisnost obeh komponent, je ocena skupne variance $\hat{\sigma}_\varepsilon^2(\vec{x})$ vsota: $\hat{\sigma}_m^2(\vec{x}) + \hat{\sigma}_p^2(\vec{x})$. Ob normalni porazdelitvi šuma, je napovedni interval definiran kot:

$$\hat{y}(\vec{x}) \pm z_{\alpha/2}\, \hat{\sigma}_\varepsilon^2(\vec{x}) \tag{3.9}$$

kjer je $\hat{y}(\vec{x})$ napoved osnovnega modela in z vrednost standardizirane spremenljivke. Avtorji metode so za računanje napovednih intervalov predlagali Studentovo t-porazdelitev, ki je primernejša, kadar je statistika računana na manjšem vzorcu. Mi zaradi primerljivosti metod uporabimo standardno z-vrednost za izbrano vrednost α.

Velika časovna zahtevnost metode je najbolj pogojena z učenjem modela bagging. Za b vzorčenj potrebujemo čas reda $O(b \cdot (M_L + n \cdot M_P))$. Iz n učnih primerov dobimo n prirejenih kvadratov residualov, kar zaradi popravkov skupne napovedi zahteva dodaten čas reda $O(b \cdot n \cdot M_P)$. Za napovedovanje variance šuma podatkov naučimo nov model. V ta namen smo uporabili model RBFN, ki za učenje potrebuje čas reda $O(n^2)$ in $O(M_P + n + 1)$ za izračun napovednega intervala.

Implementirali smo dve različici. Poenostavljena BML-A pri računanju vrednosti r^2 ne upošteva varianco negotovosti primera za posamezen primer. Različica BML-A* je implementirana, kot jo definira avtor metode.

3.1.3.2 Zapranis & Livanis (BML-B)

Enostavnejša različica zgoraj omenjene metode BML-A je bila objavljena v delu [41]. Od originalne metode se razlikuje v dveh podrobnostih.

Varianco negotovosti modela prav tako podamo z varianco napovedi notranjih modelov modela bagging. Prva razlika je pri pridobivanju množice za učenje napovedovanja variance šuma

podatkov. Pri računanju razlik ne razlikujemo med posameznimi primeri in vedno odštejemo skupno napoved vseh notranjih modelov. Hkrati pri ciljnih vrednostih ne upoštevamo varianco negotovosti modela:

$$r^2(\vec{x}) = (y(\vec{x}) - \hat{y}_{bag}(\vec{x}))^2 \tag{3.10}$$

$$\hat{y}_{bag}(\vec{x}) = \frac{1}{b}\sum_{i=1}^{b} \hat{y}_i(\vec{x}) \tag{3.11}$$

Druga razlika je v srednji vrednosti napovednega intervala. Sredino intervala ne predstavlja napoved osnovnega modela ampak napoved modela bagging. Predpostavka je, da ta nudi boljšo oceno prave vrednosti regresijske funkcije. Intervali zaupanja so tako podani kot:

$$\hat{y}_{bag}(\vec{x}) \pm z_{\alpha/2}\, \hat{\sigma}_{\varepsilon}^2(\vec{x}) \tag{3.12}$$

Tudi tukaj računanje napovednih intervalov posplošimo in uporabimo standardno z-vrednost.

Časovna zahtevnost metode BML-B je enaka originalni metodi. Razlikuje se le v členu $O(b \cdot n \cdot M_P)$, ki odpade zaradi poenostavljenega računanja kvadratov residualov.

3.1.4 Lokalne okolice

Preprostejše in časovno učinkovitejše metode delujejo na podlagi preiskovanja lokalne okolice. Nasprotno od metod stremljenja in maksimalnega verjetja poskušajo skupno varianco napovedi oceniti neposredno.

3.1.4.1 Najbližji sosedi (NN-5 in NN-100)

Za posamezen primer lahko lokalno zanesljivost njegove napovedi ocenimo s pomočjo njegovih najbližjih sosedov. Predpostavka je, da so prave vrednosti novih primerov porazdeljene enako kot za že znane primere v bližnji okolici.

Za vse primere učne množice izračunamo njihove predznačene residuale. Ob predpostavki, da so ti normalno porazdeljeni, nam njihova varianca predstavlja oceno skupne variance napovedi. Z metodo najbližjih sosedov so napovedni intervali podani kot:

$$\hat{y}(\vec{x}) + \bar{r} \pm z_{\alpha/2}\, \hat{\sigma}_{\varepsilon}^2(\vec{x}) \tag{3.13}$$

kjer je sredina intervala popravljena z vrednostjo $\bar{r}$, ki je povprečje residualov primerov v lokalni okolici. Popravek služi za odpravljanje pristranskosti modela.

Število upoštevanih najbližjih sosedov je podano kot parameter metode. Preizkusili smo dve vrednosti, in sicer 5, kjer upoštevamo 5 odstotkov najbližjih primerov učne množice ter 100, kjer je $\bar{r}$ povprečje celotne množice residualov. Inačici metode ustrezno poimenujemo z NN-5 in NN-100.

Časovno najzahtevnejše je iskanje najbližjih sosedov, ki potrebuje čas reda $O(n \cdot m \cdot \log n)$. Če pri računanju popravka upoštevamo celotno množico residualov, je metoda računsko še manj zahtevna.

Velja omeniti, da je uspešnost metode zelo odvisna od najdene lokalne okolice. Če učna množica ni dobro uravnotežena ali je model pristranski, lahko napovedni interval ne vključuje napovedi osnovnega modela.

3.1.4.2 Razvrščanje v skupine (CL)

Učinkovita metoda, ki za ocenjevanje zanesljivosti uporablja idejo razvrščanja v skupine, je bila objavljena v delu [36]. Učni podatki so razdeljeni v mehke množice, glede na velikost napake njihovih napovedi. Za vsako skupino posebej se tvori napovedni interval, na katerih se nauči model za ocenjevanje zanesljivosti novih primerov.

Preprostejša različica omenjene metode je predstavljena v [28]. Ta uporablja klasično razvrščanje v skupine (angl. *clustering*) z definiranjem k centroidov. Izbrano število skupin je odvisno od velikosti učne množice in je podano hevristično s $k = \lceil \sqrt{n/2} \rceil$.

Za vsako gručo posebej se napovedni intervali tvorijo z upoštevanjem empirične distribucije residualov. $(1 - \alpha)$ napovedni interval podajata $(100 \cdot \alpha/2)$. in $(100 - 100 \cdot \alpha/2)$. percentil skupine residualov. Intervalna ocena zanesljivosti novega primera po metodi CL je določena z napovednim intervalom gruče, v katero je primer razvrščen. Posledično lahko metoda spremeni napoved osnovnega modela (določa srednjo vrednost intervala).

Časovna zahtevnost uporabljenega algoritma za razvrščanje v skupine K-means [17] je $O(n^{5/2} \cdot m)$. Za tvorjenje intervalov zaupanja je potrebno residuale znotraj posameznih gruč urediti po velikosti, kar zahteva dodaten čas reda $O(\sqrt{n/2} \cdot \sqrt{2n} \cdot \log \sqrt{2n})$. Časovna zahtevnost iskanja skupine novega primera je $O(m \cdot \log \sqrt{n/2})$.

3.1.4.3 Kvantilni regresijski gozdovi (QRF)

Intervalno oceno zanesljivosti lahko podamo tudi na podlagi adaptivnih lokalnih okolic. Metoda kvantilni regresijski gozdovi (angl. *quantile regression forest*) [24] se od naključnih gozdov razlikuje v tem, da listi dreves ohranijo vse primere iz učne množice. V delu [23] predstavljena metoda izkorišča dodatno informacijo za ocenjevanje pogojnih mejnih vrednosti oz. kvantilov, ki jo lahko uporabimo za učenje residualov.

Pogojna funkcija porazdelitve residualov r, je definirana kot:

$$F(r|X = \vec{x}) = P(R \leq r|X = \vec{x}) = E(1_{\{R \leq r\}}|X = \vec{x}) \tag{3.14}$$

Vrednost funkcije lahko ocenimo z uteženim povprečjem:

$$\hat{F}(r|X = \vec{x}) = \sum_{i=1}^{b} w_i(\vec{x}) \cdot 1_{\{R \leq r\}} \tag{3.15}$$

kjer so w_i ($\sum w_i = 1$) uteži posameznih dreves, relativne glede na število primerov v listih ter $1_{\{R \leq r\}}$ predstavlja indikatorsko spremenljivko:

$$1_{\{R \leq r\}} = \begin{cases} 1 & R_i \leq r \\ 0 & sicer \end{cases} \tag{3.16}$$

Pogojne kvantile ocenimo z največjo vrednostjo r, za katero velja $\hat{F}(r|X = \vec{x}) \geq \alpha$.

Ko prispe novi primer, najprej izračunamo uteži, tako da ga spustimo po drevesih gozda, kjer so residuali r_i v listih urejeni po naraščajočih vrednostih. Zgornjo in spodnjo mejo $(1 - \alpha)$ napovednega intervala določata indeksa *s* in *z*, pri katerih je $\sum_{i=1}^{s} w_i \geq \alpha/2$ in $\sum_{i=1}^{z} w_i \geq 1 - \alpha/2$. Intervalna ocena zanesljivosti metode QRF je podana kot:

$$[\hat{y}(\vec{x}) + r_s\ , \hat{y}(\vec{x}) + r_z] \tag{3.17}$$

Gradnja gozda z b drevesi zahteva čas reda $O(b \cdot n \cdot m \cdot \log n)$ ter iskanje v drevesih $O(b \cdot \log n)$. Pri podajanju napovedi moramo preiskati vsa drevesa za kar je potreben čas $O(b \cdot \log n)$, izračun napovednega intervala pa traja konstanten čas.

Poglavje 4 Opis problema in podatkov

V tem poglavju predstavimo uporabljene testne množice regresijskim problemov in navedemo mere in druge pristope, s katerimi vrednotimo uspešnost metod intervalnega ocenjevanja zanesljivosti. Opišemo tudi testno okolje in potek testiranja za analizo metod.

4.1 Testne množice

Metode za intervalno ocenjevanje zanesljivosti smo preizkusili na 22 različnih (20 realnih in 2 umetnih) podatkovnih množicah za regresijsko napovedovanje, ki smo jih zbrali iz repositorijev *UCI* [5], *DELVE* [31] in *StatLib* [40] ter spletne strani *LIACC* [38]. Pri podatkovnih tokovih navadno govorimo o preprostih podatkih, zato smo iskali množice z največ 20 atributi. Premajhne množice smo med testiranjem z različnimi permutacijami združili v večje. Primere z manjkajočimi podatki smo pri obravnavi izpustili.

Dodatno smo generirali 20 množic z umetnimi problemi. Podatki predstavljajo linearno, linearno odsekovno (lomljeno), odsekoma konstantno in nelinearno funkcijo. Za najenostavnejšo linearno funkcijo generiramo po dve množici z dodanim homogenim in heterogenim šumom (linearno odvisnim od neodvisne spremenljivke). Problem s heterogenim šumom otežimo tudi pri nelinearni funkciji. Pri odsekovni in odsekoma konstantni funkciji vrednosti pokvarimo s šumom, vzorčenim iz različnih Gaussovih porazdelitev. Vsak problem dodatno razširimo v funkcije dveh, treh in štirih neodvisnih spremenljivk. Množice generirane s funkcijo ene spremenljivke uporabimo za grafično analizo intervalnih cenilk. Na kompleksnejših problemih z več spremenljivkami izvajamo validacijo točnosti napovednih algoritmov. Vsaka umetna množica vsebuje 2000 vzorčenih primerov v naključnem vrstnem redu.

4.2 Mere uspešnosti

Za ocenjevanje uspešnosti ocene zanesljivosti lahko uporabimo različne mere. Pravilnost napovednega intervala ocenimo z mero, ki jo imenujemo *pokrivna verjetnost napovednih intervalov* (angl. *prediction interval coverage probability*) in predstavlja verjetnost, da je prava

vrednost odvisne spremenljivke zajeta znotraj napovednega intervala. Oceno izračunamo kot:

$$PVNI = \frac{1}{n}\sum_{i=1}^{n} c_i \tag{4.1}$$

$$c_i = \begin{cases} 1 & y_i \epsilon [I_i^S, I_i^Z] \\ 0 & sicer \end{cases}$$

kjer je n število primerov in ima c_i vrednost 1, če je prava vrednost i-tega primera vsebovana v njegovem napovednem intervalu s spodnjo mejo I^S in zgornjo I^Z. Za pravilne napovedne intervale bo izračunana vrednost blizu zastavljene verjetnosti $1 - \alpha$.

Prav tako je pomembna širina napovednih intervalov, kot mera njegove optimalnosti. *Povprečni napovedni interval* (angl. *mean prediction interval*) meri povprečno širino intervala:

$$PNI = \frac{1}{n}\sum_{i=1}^{n} (I_i^Z - I_i^S) \tag{4.2}$$

Ker želimo metode med seboj primerjani na različnih problemskih domenah, povprečni napovedni interval normaliziramo s *privzetim napovednim intervalom*, ki ga dobimo iz empirične porazdelitve odvisne spremenljivke na celotni učni množici. Tako ocenimo *relativni povprečni napovedni interval* (angl. *relative mean prediction interval*):

$$RPNI = \frac{1}{n}\sum_{i=1}^{n} \frac{(I_i^Z - I_i^S)}{y_{(1-\alpha/2)} - y_{\alpha/2}} \tag{4.3}$$

$y_{(1-\alpha/2)}$ in $y_{\alpha/2}$ predstavljata ustrezna percentila opazovanih vrednosti odvisne spremenljivke.

Zaželeni so napovedni intervali, katerih pokrivna verjetnost je čim bližje podani verjetnosti $1 - \alpha$ in je vrednost relativnega povprečnega napovednega intervala čim manjša. Oceni uspešnosti se med seboj izključujeta. Pravilnejši napovedni intervali bodo praviloma širši, medtem kot za optimalnejše napovedne intervale pričakujemo manjšo pokrivno verjetnost. V [28] je predlagana kombinirana ocena, ki združuje obe vrednosti. *Kombinirana statistika* je definirana kot:

$$RPNI\text{-}PVNI = 100 \cdot RPNI + \log(\max((1-\alpha) - PVNI)^2, 10^{-10})) \tag{4.4}$$

Iz enačbe vidimo, da ta daje večji pomen ocenjeni vrednosti RPNI in je doprinos logaritma na račun ocene PVNI omejen.

Pri modeliranju podatkovnih tokov je čas izvajanja zelo pomemben. Primerjali smo porabo časa različnih algoritmov učenja z in brez uporabe ocen zanesljivosti. Dobra ocena uporabnosti metode je faktor upočasnitve, ki ga podamo kot:

$$S = \frac{\overline{T_{zNI}}}{\overline{T_{brezNI}}} \tag{4.5}$$

V enačbi $\overline{T_{brezNI}}$ in $\overline{T_{zNI}}$ predstavljata povprečna časa, ki ga porabita osnovni in nadgrajeni učni algoritem za izgradnjo modela ter podajanje individualne napovedi. Za metodo želimo, da je njen faktor upočasnitve čim manjši.

Časovna zahtevnost metode vpliva na njeno učinkovitost. V enakem času lahko prirejeni učni algoritem obdela manj učnih primerov in posledično so njegove napovedi lahko slabše. Zanimiva je primerjava doseženih točnosti obeh izvedb algoritma pri enakih časovnih omejitvah (dosežemo z zmanjšanjem velikosti okna in s tem učne množice). Merilo točnosti regresijskega modela je *koren srednje kvadratne napake* napovedi (angl. *root mean squared error*), ki ga izračunamo kot:

$$RMSE = \sqrt{\frac{\sum_{i=1}^{N}(\hat{y}_i - y_i)^2}{n}} \tag{4.6}$$

kjer je $\hat{y}_i$ napoved modela, y_i prava vrednost in n število napovedanih primerov. Za primerjavo točnosti med različnimi modeli na različnih problemskih množicah izračunamo *normaliziran koren srednje kvadratne napake* (angl. *normalized root mean squared error*):

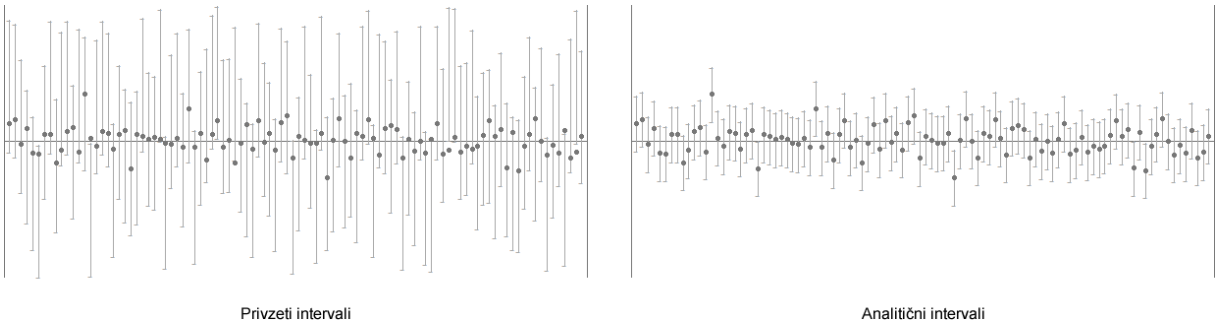
$$NRMSE = \frac{RMSE}{y_{max} - y_{min}} \tag{4.7}$$

y_{max} in y_{min} predstavljata največjo in najmanjšo vrednost odvisne spremenljivke opazovanih primerov.

4.3 Vizualizacije rezultatov

O delovanju različnih metod lahko sklepamo iz grafičnih predstavitev intervalnih ocen zanesljivosti. S primerno vizualizacijo podatkov lahko primerjamo različne pristope in njihovo uspešnost.

Preprosta vizualna tehnika primerjave napovednih intervalov več metod je predstavljena v delu [24]. V podatkovnem toku je vrstni red podatkov točno določen, običajno pa primeri niso urejeni po vrednosti atributov. Prihaja lahko do večjih skokov v vrednosti odvisnih spremenljivk. Hkrati je točnost napovedi neposredno odvisna od predhodno videnih primerov. Za primerjavo uspešnosti metod na podatkovnih tokovih je zato bolj primerna vizualizacija, ki prikazuje napake napovedi (residuale) in meje napovednih intervalov glede na pravo vrednost odvisne spremenljivke [29]. Abscisna os x predstavlja opazovane ciljne vrednosti. V smeri ordinatne osi y pa so podane vrednosti residualov osnovnih napovedi in odstopanja mej napovednih intervalov. Na sliki 4.1 je prikazan primer za najenostavnejše konstantne in analitične napovedne intervale.



Slika 4.1:

Prikaz privzetih in analitičnih napovednih intervalov modela linearna regresija na problemu enostavne linearne funkcije ene spremenljivke.

4.4 Testno okolje

Metode smo implementirali in testirali v programskem okolju R. R je popularen odprtokodni programski jezik za analizo podatkov in statistiko. V okolju R najdemo implementacije številnih statističnih orodij in metod strojnega učenja, med drugim tudi vse omenjene regresijske napovedne modele.

Izbiramo lahko med številnimi paketi, ki ponujajo rešitve. Uporabili smo naslednje implementacije: `knn.reg` iz paketa `FNN` za algoritem kNN, za linearno regresijo in posplošeni linearni model funkciji `lr` ter `glm` paketa `stats`, ki je del osnovne distribucije, paket `rpart` za odločitvena drevesa, model naključnih gozdov iz paketa `randomForest`, realizacijo modela SVM iz paketa `e1071` in paket `nnet` za učenje umetnih nevronskih mrež z enim skritim nivojem. Pri klicih vseh funkcij smo uporabili privzete parametre.

Metode intervalnega ocenjevanja zanesljivosti smo implementirali sami. Pri tem smo za napovedovanje s kvantilnimi regresijskimi gozdovi uporabili paket `quantregForest`. Model mreže radialnim baznih funkcij in metoda bagging sta enostavnejša in smo ju realizirali sami.

Teste smo izvajali na računalniku z mobilnim procesorjem Intel Core i7, ki deluje pri taktu 2,8 GHz. Izvajalne čase algoritmov smo beležili z R-funkcijo `get_nanotime` iz paketa `microbenchmark`, ki meri čas v nanosekundah. Meritve smo zaokrožili na mikrosekunde.

4.5 Metodologija testiranja

Podatkovni tok simuliramo tako, da primere zaporedno beremo iz množice podatkov. Pri tem drseče okno postopoma pomikamo po primerih in na učni množici, ki jo definira okno, poženemo učne algoritme regresijskih modelov ter metode za ocenjevanje zanesljivosti. Nove modele nato uporabimo za napovedovanje primerov, ki sledijo. Intervalne ocene podamo v obliki 95% napovednih intervalov. Postopek je ponovljen z okni različnih velikosti, in sicer 20, 50, 100, 250 in 500 primerov.

Najbolj direkten, vendar tudi časovno najbolj potraten pristop je, da položaj okna premaknemo vsakič, ko podamo eno samo napoved. Uspešnost algoritmov je v tem primeru zelo odvisna od vrstnega reda branja primerov, zato smo se odločili za drugačen pristop. Z novim modelom napovemo $1/3 \cdot w$ (w je velikost okna) primerov in za isti korak pomikamo okno. Tako dobimo manj pristranske regresijske napovedi in ocene zanesljivosti.

Za relevantnost rezultatov želimo, da se drseče okno premakne vsaj 5 krat. Pri večjem oknu to lahko pomeni, da imamo v nekaterih problemskih množicah premalo podatkov. V tem primeru različne permutacije majhne množice združimo v večjo. Pri tem ne dovolimo, da isti primer nastopi preblizu skupaj, kar bi povzročilo, da se pri pomiku okna v učni množici pojavi večkrat. Za stabilnost rezultatov celoten test ponovimo 5 krat z različnim vrstnim redom podatkov (celotne množice permutiramo).

Iz posameznih napovedi osnovnega modela ocenimo točnost napovedovanja. Za ocene zanesljivosti različnih metod so izračunane pokrivne verjetnosti, relativni napovedni interval in kombinirana statistika. Privzeti napovedni interval izračunamo za vsako učno množico posebej, torej pri vsakem premiku okna. Zabeleženi so tudi povprečni izvajalni časi učenja in napovedovanja algoritmov z in brez ocen zanesljivosti.

Velja opozoriti, da se pri poskusih ne ukvarjamo z optimizacijo posameznih modelov. Uporabimo privzete vrednosti parametrov in nastavitve implementacij modelov v okolju R. Posledično se lahko nekateri napovedni modeli obnesejo slabše kot sicer.

Poglavje 5 Rezultati

Predstavljene metode intervalnega ocenjevanja zanesljivosti smo implementirali in preizkusili z različnimi regresijskimi napovednimi modeli. Glavne rezultate testiranj prikažemo v nadaljevanju.

5.1 Pravilnost in optimalnost intervalov

Rezultate intervalnih cenilk podamo v obliki napovednih intervalov. Izbrana je stopnja tveganja $\alpha = 0{,}05$, torej nas zanimajo 95% napovedni intervali. Rezultati na realnih podatkovnih množicah nam dajo porazdelitev vrednosti PVNI. Najboljša je tista metoda, ki v povprečju doseže PVNI najbližji vrednosti 0,95 in ima čim manjši raztros. Rezultati testiranj so prikazani v tabeli 5.1.

50	ML	BML-A	BML-A*	BML-B	NN-5	NN-100	CL	QRF
$PVNI_{povp}$	0,778	0,963	**0,957**	0,921	0,644	0,786	0,654	0,710
$PVNI_{2,5}$	0,384	0,88	0,872	0,772	0,294	0,413	0,363	0,394
$PVNI_{97,5}$	0,963	0,994	0,991	0,994	0,834	0,944	0,806	0,872

100	ML	BML-A	BML-A*	BML-B	NN-5	NN-100	CL	QRF
$PVNI_{povp}$	0,842	0,965	**0,960**	0,931	0,757	0,837	0,737	0,793
$PVNI_{2,5}$	0,427	0,894	0,891	0,782	0,418	0,421	0,412	0,427
$PVNI_{97,5}$	0,973	0,997	0,994	0,997	0,912	0,958	0,864	0,918

500	ML	BML-A	BML-A*	BML-B	NN-5	NN-100	CL	QRF
$PVNI_{povp}$	0,911	0,969	0,965	**0,945**	0,875	0,895	0,877	0,895
$PVNI_{2,5}$	0,575	0,853	0,845	0,735	0,560	0,617	0,581	0,587
$PVNI_{97,5}$	1,000	1,000	1,000	1,000	0,987	0,986	0,994	0,996

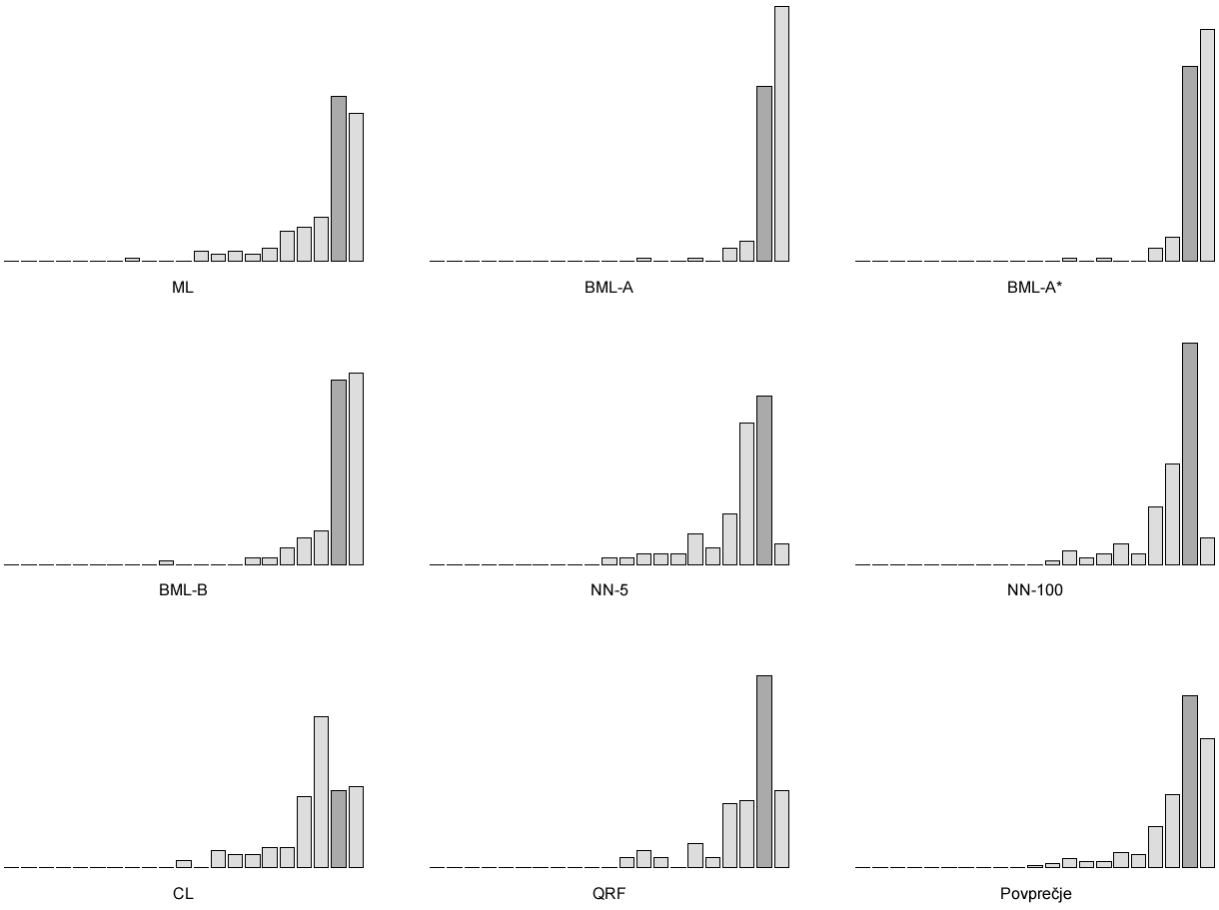
Tabela 5.1:

Dosežene vrednosti PVNI intervalnih cenilk. V stolpcih PVNI so podani srednja vrednost ter 2,5. in 97,5. percentil povprečne porazdelitve vrednosti PVNI doseženih z vsemi modeli na realnih podatkovnih množicah pri različnih velikostih drsečega okna.

Po obeh kriterijih (najbližja srednja in najmanjši raztros vrednosti PVNI) dajo metode s pristopom stremljenja najboljše rezultate. Njihovi napovedni intervali so pravilnejši. Metode

lokalnih okolic podcenjuje srednjo vrednost in razpon vrednosti PVNI je pri njih večji. Presenetljivo dobro se izkaže metoda ML. Čeprav ne upošteva variance negotovosti modela, dosega rezultate, ki so primerljivi ali celo boljši od metod na osnovi lokalnih okolic.

Z večjim drsečim oknom metode dosegajo boljše rezultate. Izjemi sta različici BML-A, ki precenjujeta srednjo vrednost PVNI. Pri vseh se raztros vrednosti pokrivnih verjetnosti zmanjša. Porazdelitve doseženih vrednosti PVNI različnih metod prikazuje slika 5.1.



Slika 5.1:

Porazdelitve doseženih vrednostih PVNI na realnih testnih množicah pri oknu velikosti 500 primerov. Gostote so izračunane v korakih po 0,50. S temnejšo barvo so obarvani stolpci, ki predstavljajo zastavljeno vrednost 0,95. Zadnji histogram prikazuje povprečne gostote vrednosti PVNI za vse metode skupaj.

Vidimo, da je pri vseh metodah vrh gostote vrednosti PVNI blizu želeni. Največ pravilnih (zaokroženih) vrednosti dosežejo metode najbližjih sosedov in kvantilni regresijski gozd, vendar imajo napram metodam BML večji raztros. Tudi s primerjavo histogramov bi zaključili, da metode BML podajo najbolj pravilne napovedne intervale. Povprečna porazdelitev vseh metod skupaj, ki jo prikazuje graf desno spodaj, je vizualno najbližje prikazu metode ML.

Na dosežene vrednosti PVNI neposredno vplivajo širine napovednih intervalov. Praviloma ožji intervali zajamejo manj pravih vrednosti, kar se kaže v nizkih vrednostih PVNI. Tabela 5.2 prikazuje rezultate metod intervalnega ocenjevanja zanesljivosti glede na različne mere uspešnosti.

50	ML	BML-A	BML-A*	BML-B	NN-5	NN-100	CL	QRF
PVNI	0,778	0,963	**0,957**	0,921	0,644	0,786	0,654	0,710
RPNI	0,384	0,999	0,886	0,721	**0,318**	0,494	0,360	0,384
PVNI-RPNI	0,963	99,967	88,619	72,140	**31,794**	49,389	36,016	38,416

100	ML	BML-A	BML-A*	BML-B	NN-5	NN-100	CL	QRF
PVNI	0,842	0,965	**0,960**	0,931	0,757	0,837	0,737	0,793
RPNI	0,487	0,814	0,765	0,628	**0,352**	0,504	0,377	0,387
PVNI-RPNI	48,679	81,389	76,530	62,795	**35,232**	50,416	37,655	38,692

500	ML	BML-A	BML-A*	BML-B	NN-5	NN-100	CL	QRF
PVNI	0,911	0,969	0,965	**0,945**	0,875	0,895	0,877	0,895
RPNI	0,497	0,642	0,623	0,542	**0,400**	0,574	0,420	0,416
PVNI-RPNI	49,752	63,829	62,330	54,170	**39,972**	57,416	42,039	41,562

Tabela 5.2:

Ovrednotenje intervalnih cenilk na testnih podatkovnih množicah. Po stolpcih so podani izračunane povprečne vrednosti PVNI, RPNI in kombinirane statistike pri različnih velikostih učne množice za vse regresijske modele na realnih podatkovnih množicah.

Metode stremljenja, ki dosegajo najvišje vrednosti PVNI, tvorijo najširše napovedne intervale (višje vrednosti RPNI). Posledično so po kriteriju skupne statistike ocenjene najslabše. Dobre rezultate dajo metode na osnovi lokalnih okolic. V povprečju so njihovi napovedni intervali optimalnejši in dosežejo nižje vrednosti PVNI-RPNI. Rezultati metode ML so primerljivi z NN-100, ki je s skupno statistiko najslabše ocenjena med intervalnimi cenilkami s pristopom lokalnih okolic. V povprečju najbolj točne napovedne intervale (najnižja ocena skupne statistike) poda metoda NN-5.

Zanimivo je, da so rezultati skupne statistike za večino metod najslabši na največjem oknu. Njihovi napovedni intervali so pravilnejši, vendar tudi širši. Nasprotno pa metode BML pri večji učni množici podajo bolj optimalne intervale.

5.2 Časovna zahtevnost

Za primerjavo najprej analiziramo izvajalne čase osnovnih regresijskih modelov brez ocen zanesljivosti pri različnih velikostih drsečega okna na 20 realnih testnih množicah. Rezultati meritev so prikazani v tabeli 5.3.

	kNN	LR	GLM	RT	RF	SVM	ANN
50	124	2.792	3.423	5.930	25.736	5.306	11.186
	793	1.480	1.535	2.109	2.359	1.651	1.901
	917	4.272	4.958	8.039	**28.095**	6.957	13.087
100	138	2.809	3.418	6.890	61.457	6.910	21.963
	887	1.438	1.516	2.173	2.864	1.638	1.939
	1.025	4.247	4.934	9.063	**64.321**	8.548	23.902
500	142	3.431	4.519	14.751	507.038	55.749	85.366
	1.906	1.797	1.563	2.504	11.593	1.993	2.017
	2.048	5.228	6.082	17.255	**518.631**	57.742	87.383

Tabela 5.3:

Povprečni izvajalni časi regresijskih učnih algoritmov. Posebej je izmerjen čas učenja modela in napovedovanje individualnega primera. Seštevek prestavlja skupen izvajalni čas. Rezultati so podani v mikrosekundah (μs).

Iz meritev je očitno, da je izvajalni čas algoritmov odvisen od števila učnih primerov in velikosti problema (število atributov). Rezultati kažejo na eksponentno rast, pri čemer je stopnja rasti odvisna od modela. Najhitreje podajo svoje napovedi preprostejši regresijski modeli (kNN, LR, GLM in RT). V naših poskusih se je kot časovno najzahtevnejši izkazal model naključni gozdovi. Z drugačnimi parametri pri izvajanju algoritmov bi lahko bil vrstni red drugačen.

Napovedovanje z oceno zanesljivosti upočasni učni algoritem. Koliko, je odvisno od uporabljene metode intervalnega ocenjevanja in regresijskega modela. V tabeli 5.4 so prikazani izvajalni časi metod intervalnega ocenjevanja pri napovedovanju s časovno najmanj zahtevnim modelom kNN. Podajanje napovednih intervalov je bistveno bolj potratno za časovno zahtevnejše modele. Meritve za algoritem naključni gozdovi so prikazane v tabeli 5.5.

	ML	BML-A	BML-A*	BML-B	NN-5	NN-100	CL	QRF
50	8.797	46.441	47.481	45.032	959	969	3.239	18.171
	4.939	23.621	23.819	23.847	4.243	27	1.113	2.553
	13.736	70.062	71.300	**68.879**	5.202	**996**	4.352	20.724
100	14.814	57.401	58.811	63.370	1.343	1.321	4.950	23.792
	6.404	26.670	27.000	27.232	5.680	27	1.066	2.394
	21.218	84.071	85.811	**90.602**	7.023	**1.348**	6.016	26.186
500	136.238	249.361	254.330	313.313	5.897	5.909	22.151	64.554
	51.549	91.533	96.767	92.509	50.354	29	3.514	2.669
	187.787	340.894	351.097	**405.822**	56.251	**5.938**	25.665	67.223

Tabela 5.4:

Izvajalni časi metod intervalnega ocenjevanja zanesljivosti za model kNN. Posebej so podani časi za učenje intervalnih cenilk, tvorjenje individualnega napovednega intervala in skupni izvajalni čas. Časovne enote so mikrosekunde (µs).

	ML	BML-A	BML-A*	BML-B	NN-5	NN-100	CL	QRF
50	11.191	861.323	864.274	871.795	3.457	3.478	5.642	20.852
	4.839	76.688	74.720	73.766	4.153	28	1.074	2.506
	16.030	938.011	938.994	**945.561**	7.610	**3.507**	6.716	23.358
100	18.503	1.923.764	1.916.747	1.956.689	5.263	5.273	8.996	28.313
	6.335	89.609	89.156	88.760	5.713	28	1.135	2.949
	24.838	2.013.373	2.005.903	**2.045.449**	10.976	**5.301**	10.131	31.262
500	141.085	15.196.190	14.937.179	15.240.576	26.146	26.151	41.629	91.681
	36.489	228.518	236.331	235.006	36.034	29	3.495	2.371
	177.574	15.424.708	15.173.510	**15.475.582**	62.181	**26.180**	45.124	94.052

Tabela 5.5:

Izvajalni časi intervalnih cenilk za model naključni gozdovi. Posebej so podani časi za učenje intervalnih cenilk, tvorjenje individualnega napovednega intervala in skupni izvajalni čas. Časovne enote so mikrosekunde (µs).

Izvajalni časi metod intervalnega ocenjevanja zanesljivosti so prav tako eksponentno odvisni od velikosti učne množice. Na porabljen čas pa vpliva tudi izbira učnega algoritma. Pri metodah ML, NN-5, NN-100, CL in QRF se pri učenju uporabljajo residuali, ki se izračunajo iz napovedi osnovnega modela. Kot lahko vidimo iz rezultatov v tabeli 5.3, kompleksnejši algoritmi za podajanje napovedi potrebujejo več časa. Bistveno bolj je vpliv izbire napovednega modela očiten pri metodah BML. Učenje se ponovi za vsak nov notranji model modela bagging, kar pripelje do še večjih upočasnitev algoritma.

Izračunani faktorji upočasnitve regresijskih modelov z metodami intervalnega ocenjevanja so podani v tabeli 5.6. Pri računanju povprečij nismo upoštevali meritve za model kNN. Ker ta ne pozna pravega učenja, ga obravnavamo posebej.

	ML	BML-A	BML-A*	BML-B	NN-5	NN-100	CL	QRF
50	3,0	43,3	43,2	40,0	1,4	1,4	1,8	4,9
	3,7	32,6	32,3	31,2	3,4	1,0	1,6	2,4
	3,1	**40,1**	39,9	37,5	1,9	**1,3**	1,8	4,1
100	3,8	43,1	43,2	40,7	1,3	1,3	2,0	5,4
	4,5	33,2	33,0	31,7	4,2	1,0	1,6	2,4
	3,8	40,2	**40,3**	38,4	2,0	**1,3**	1,9	4,5
500	14,5	50,3	51,3	47,3	1,3	1,3	2,9	7,7
	23,4	49,0	48,8	42,6	23,9	1,0	2,7	2,2
	15,0	48,5	**49,0**	45,0	5,2	**1,2**	2,7	6,2

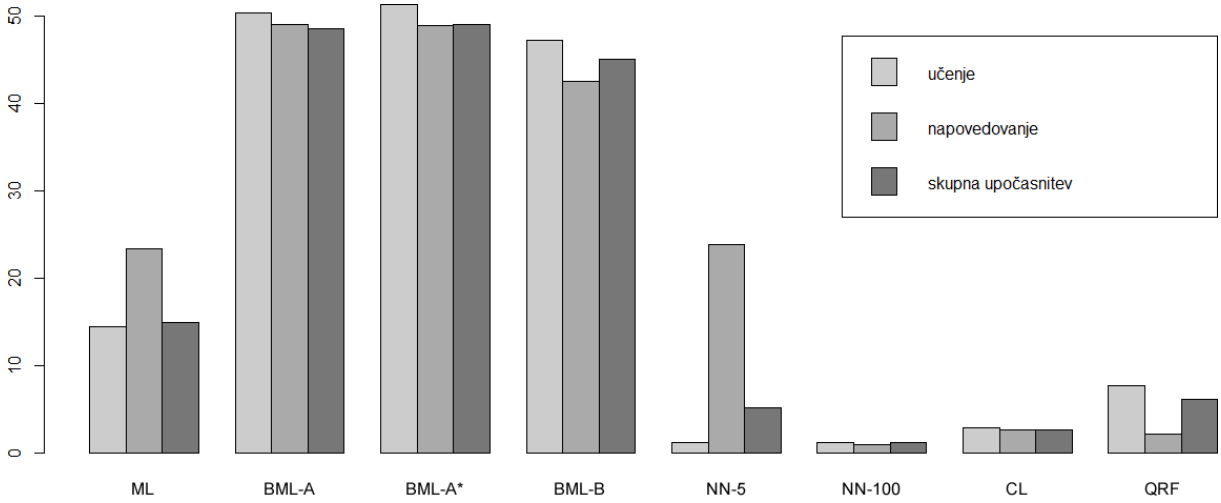
Tabela 5.6:

Faktorji upočasnitve algoritmov pri napovedovanju z oceno zanesljivosti na realnih testnih množicah. Posebej so podane upočasnitve v fazi učenja, pri napovedovanja individualnega primera in skupnega izvajalnega časa. Rezultati so povprečja vseh regresijskih modelov z izjemo algoritma kNN.

Najmanj izvajanje učnega algoritma upočasnijo metode NN-5, NN-100 in CL. Za časovno najmanj zahtevno cenilko NN-100, je opazna manjša upočasnitev le v fazi učenja modela. K skupni upočasnitvi cenilke NN-5 najbolj prispeva faza napovedovanja, saj potreben čas za iskanje najbližjih sosedov raste eksponentno. Nekoliko slabše se odrežeta metodi ML in QRF. Pri slednji je učni algoritem upočasnjen predvsem na račun gradnje dreves. Večji skok pri rezultatih z metodo ML na največjem oknu gre pripisati časovni zahtevnosti modela RBFN.

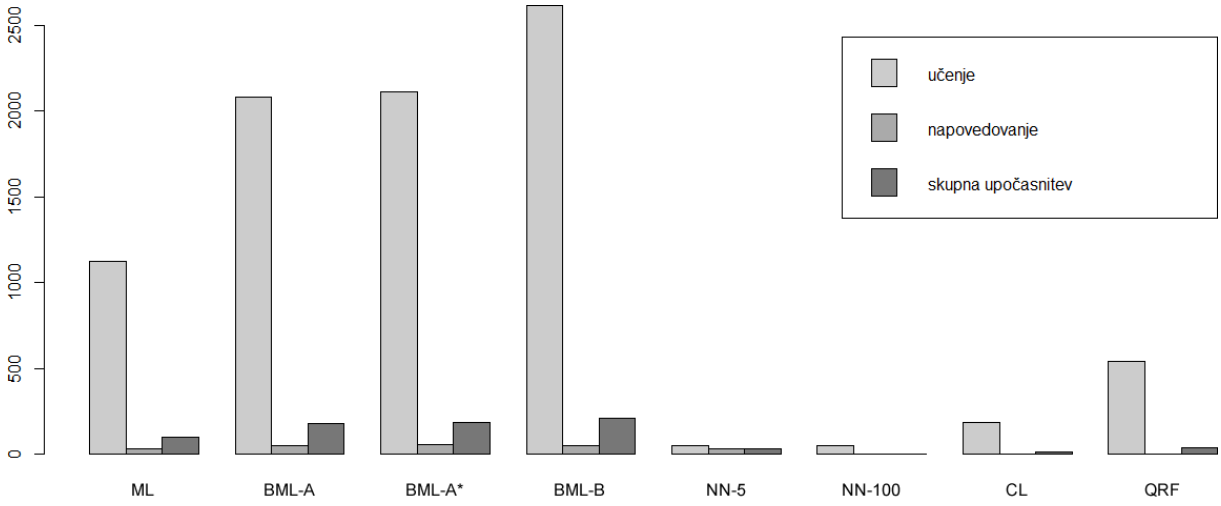
Največji faktor upočasnitve dajo intervalne cenilke, ki za oceno zanesljivosti uporabljajo pristop stremljenja. Najslabše se odrežeta različici BML-A. Za razliko od ostalih metod, je BML-B mogoče uporabiti samostojno, brez da naučimo osnovni model (interval je določen z napovedjo modela bagging). Upočasnitev je zato nekoliko manjša. Pri vseh metodah stremljenja je večji faktor upočasnitve tudi v fazi napovedovanja, kar je pri modeliranju podatkovnih tokov nezaželena lastnost.

Slika 5.2 prikazuje primerjavo faktorjev upočasnitve obravnavanih metod. Opazimo, da v povprečju metode algoritem enako upočasnijo v obeh fazah. Povsem drugače je v primeru regresijskega modela kNN. Kot vidimo na sliki 5.3, so upočasnitve bistveno večje v fazi učenja.



Slika 5.2:

Primerjava faktorjev upočasnitve metod intervalnega ocenjevanja pri največji velikosti drsečega okna. Posamezni stolpci predstavljajo upočasnitve učnih algoritmov v fazi učenja, fazi napovedovanja in skupno upočasnitev.



Slika 5.3:

Primerjava faktorjev upočasnitve algoritmov za model kNN pri oknu velikosti 500 primerov. Posamezni stolpci predstavljajo upočasnitve učnih algoritmov v fazi učenja, fazi napovedovanja in skupno upočasnitev.

5.3 Učinkovitost

Uspešnost regresijskih algoritmov podamo s točnostjo njihovih napovedi. Kot najboljši je ocenjen model, katerega srednja kvadratna napaka je najmanjša. V tabeli 5.7 so prikazane povprečne dosežene točnosti osnovnih učnih algoritmov na skupno 22 umetnih testnih množicah. Ponovno opomnimo, da se pri izvajanju testov nismo ukvarjali z optimizacijo

posameznih algoritmov in bi se lahko določeni modeli pri drugačnih parametrih in nastavitvah odrezali veliko bolje.

50	kNN	LR	GLM	RT	RF	SVM	ANN
20	0.219	0,156	0,156	**0,144**	0,160	0,203	0.473
50	0,188	0,145	0,145	**0,122**	0,135	0,172	0,234
100	0,165	0,142	0,142	**0,109**	0,119	0,152	0,149
250	0,138	0,141	0,141	**0,099**	0,111	0,134	0,103
500	0,121	0,135	0,135	0,094	0,105	0,116	**0,092**

Tabela 5.7:

Točnosti napovedi osnovnih regresijski algoritmov pri različnih velikostih drsečega okna. Podani so normalizirani koreni srednje kvadratne napake napovedi modelov povprečeni preko vseh umetnih testnih množic.

Pričakovano so točnosti napovedi modelov odvisne od velikosti učne množice in se v našem primeru izboljšujejo z večjim številom učnih primerov. V splošnem ta posplošitev seveda ne velja, saj lahko pride do prekomernega prileganja učni množici. Algoritma LR in GLM dosegata identične rezultate, kar kaže na to, da regresijsko funkcijo modelirata enako (pri izvajanju testov smo uporabili privzete parametre). Podatki kažejo tudi slabost najsodobnejšega algoritma ANN v tem, da potrebuje veliko učnih primerov, skritih nivojev/nevronov ali iteracij za podajanje dobrih napovedi.

Intervalne cenilke upočasnijo učni algoritem. Pri enakih časovnih omejitvah je ta sposoben obdelati manj učnih primerov, kar se kaže v večjih napakah njegovih napovedi. Za izhodišče vzamemo dosežene točnosti osnovnih regresijskih modelov na umetnih množicah pri največjem oknu in jih primerjamo z rezultati nadgrajenih algoritmov na manjših učnih množicah. Najmanjše drseče okno, na katerem poženemo teste, zajame 20 primerov. Točnosti regresijskih učnih algoritmov z oceno zanesljivosti so prikazane v tabeli 5.8.

	kNN	LR	GLM	RT	RF	SVM	ANN
brez	**500**	**500**	**500**	**500**	**500**	**500**	**500**
	0,121	**0,135**	**0,135**	**0,094**	**0,105**	**0,116**	**0,092**
ML	↓20	↓20	↓20	↓20	↑250	↑100	↑250
	0,219	0,156	0,156	0,144	0,160	0,152	**0,103**
BML-A	↓20	↓20	↓20	↓20	↓20	↓20	↓20
	0,219	0,156	0,156	**0,144**	0,160	0,203	0,473
BML-A*	↓20	↓20	↓20	↓20	↓20	↓20	↓20
	0,219	0,156	0,156	**0,144**	0,160	0,203	0,473
BML-B	↓20	↓20	↓20	↓20	↓20	↓20	↓20
	0,217	20,811	20,811	***0,123**	0,165	0,205	*0,238
NN-5	↓20	↓20	↓20	↓50	↓500	↓500	↓500
	*0,208	0,157	0,157	*0,125	*0,097	*0,101	**0,092**
NN-100	↑50	↑20	↑250	↓250	↓500	↓500	↓500
	*0,187	0,156	0,141	0,100	0,105	*0,115	**0,092**
CL	↓20	↓20	↓20	↑20	↓500	↑250	↓500
	0,222	0,156	0,156	0,145	0,106	*0,127	**0,098**
QRF	↓20	↓20	↓20	↓20	↓500	↑100	↓500
	*0,199	*0,147	*0,146	0,145	*0,100	*0,128	**0,092**

Tabela 5.8:

Točnosti napovedi algoritmov z oceno zanesljivosti na umetnih testnih množicah. Za vsako kombinacijo regresijskega modela in metode intervalnega ocenjevanja sta podana velikost učne množice in točnost napovedi. Pri računanju napak, je uporabljena srednja vrednost napovednih intervalov. Puščica ↓ ob velikosti oken ponazarja, da bi pri upoštevanju časovnih omejitev morali okno še zmanjšati. Pri ↑ bi okno lahko bilo večje.

Podani so rezultati, pri katerih je skupni izvajalni čas razširjenega algoritma najbližje izvajalnemu času osnovnega modela. Rezultate, ob katerih je puščica obrnjena navzgor, bi lahko z večjo učno množico še izboljšali. Puščica obrnjena navzdol pomeni, da se algoritem z oceno zanesljivosti izvaja predolgo časa in je okno potrebno nekoliko zmanjšati. Z zvezdico so označene točnosti, kjer algoritem z oceno zanesljivosti dosega boljše rezultate kot osnovni model pri enaki velikosti okna.

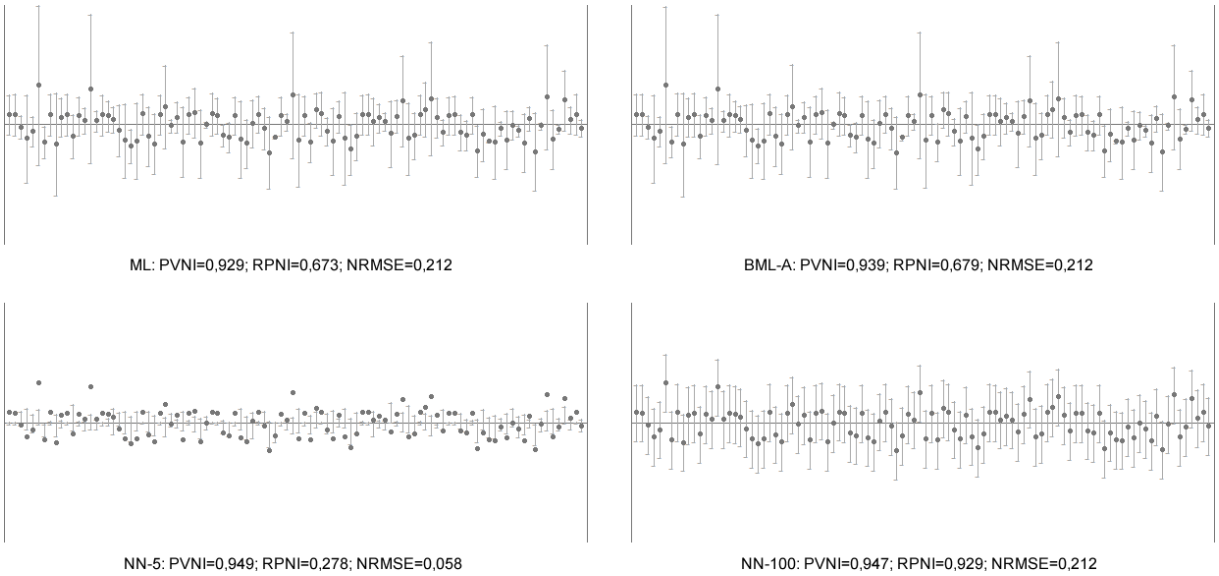
Najmanj učinkoviti so algoritmi, ki za ocenjevanje zanesljivosti uporabljajo metode BML. Tudi pri najmanjšem oknu preveč upočasnijo algoritem. Pri doslednem upoštevanju časovnih omejitev so te metode praktično neuporabne. Nekoliko bolje se odreže metoda ML. V kombinaciji z modelom RF in ANN je algoritem sposoben v enakem času obdelati več kot polovico primerov. Manjša časovna zahtevnost metod na osnovi lokalnih okolic pomeni, da so algoritmi v povprečju uspešnejši pri podajanju napovedi. Najmanjše napake napovedi dajeta obe metodi najbližjih sosedov.

Opazimo, da se nekatere dosežene točnosti razlikujejo od rezultatov osnovnih učnih algoritmov pri enaki velikosti okna (tabela 5.7). Metodi NN popravita srednjo vrednost napovednih intervalov. Prav tako napoved osnovnega modela spremenijo metode CL, QRF in BML-B. Pri metodah na osnovi lokalnih okolic to v povprečju izboljša uspešnost modelov RF in SVM, medtem ko jih pri ostalih modelih lahko tudi poslabša. Tega ne želimo posplošiti, saj so naši testi premalo obsežni. Zaradi popravka napovedi popolnoma odpove metoda BML-B v kombinaciji z modeloma LR in GLM. Vzrok je v premajhni učni množici, zaradi česa model bagging poda popolnoma drugačno napoved.

Pri rezultatih nismo upoštevali dodatne upočasnitve zaradi pogostejša premika oken. Če postopamo enako, lahko z manjšim oknom napovemo manj primerov preden okno premaknemo. Drug pristop je, da neglede na velikost okna napovemo enako število primerov, vendar bi v tem primeru napovedovali z modelom, naučenim na starejših (manj relevantnih) podatkih.

5.4 Vizualna primerjava

O lastnosti intervalnih cenilk zanesljivosti in delovanju različnih pristopov sklepamo iz grafičnih prikazov njihovih napovedi. Razlike med napovednimi intervali vidimo na sliki 5.4, ki prikazuje uspešnost nekaterih metod na umetnem nelinearnem problemu.

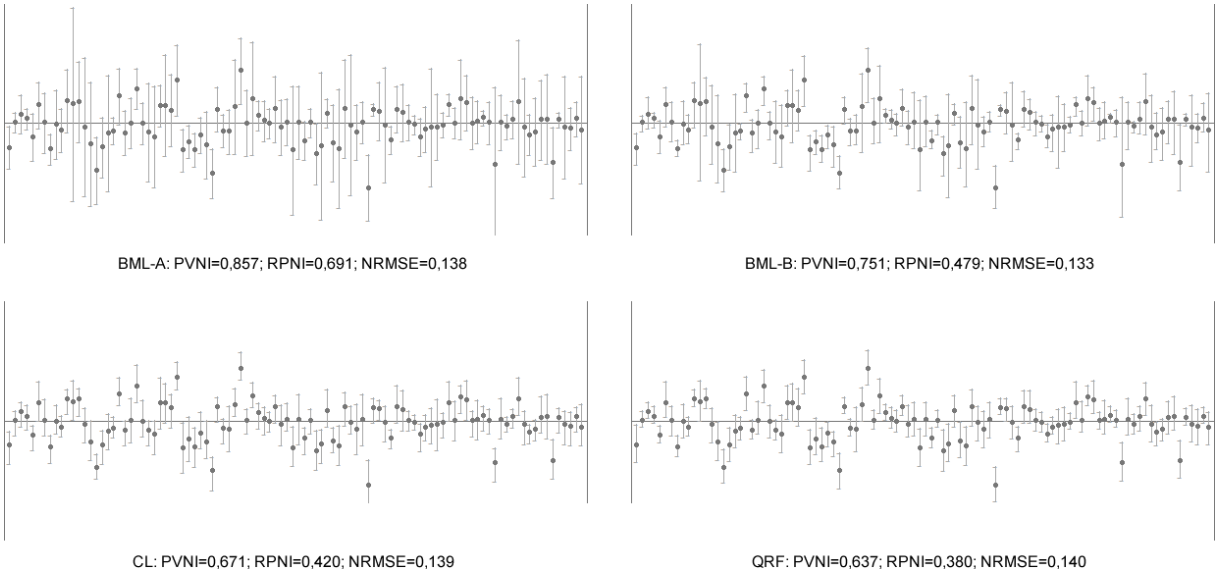


Slika 5.4:

Napovedni intervali na umetnem problemu nelinearne funkcije. Prikazih je prvih 100 napovedi metod ML, BML-A, NN-5 in NN-100 z modelom linearne regresije pri največjem drsečem oknu. Krogci predstavljajo residuale osnovnih napovedi modela in navpične črte posamezne napovedne intervale.

Na prikazih opazimo, da se napovedni intervali na osnovi maksimalnega verjetja in stremljenja zelo dobro prilegajo residualom. Intervali so širši, kadar so napovedi manj natančne in ožji, ko je napaka manjša. Pri večjih odmikih residualov so intervali celo širši od konstantnih napovednih intervalov na učni množici, kar se kaže v višji vrednosti RMPI. Bolj točni so napovedni intervali metode NN-5. Nizka vrednost RPNI se kaže na sliki z ožjimi in enakomernejšimi intervali. Na grafu desno spodaj vidimo konstantne napovedne intervale cenilke NN-100. Metoda na izbranem problemu v povprečju poda najmanj optimalne intervale, ki so le malenkost ožji kot privzeti napovedni interval. Vse predstavljene metode na izbranem preprostem nelinearnem problemu dosegajo visoko pokrivno verjetnost.

Nasprotno od metod BML, intervalne ocene na osnovi lokalnih okolic po naravi niso simetrične okoli napovedi, kar je precej očitno na grafu levo spodaj. Ocene poskušajo zajeti podatke in ne napovedi. Za naš problem nelinearne funkcije se izkaže, da napovedovanje z oceno zanesljivosti s cenilko NN-5 opazno izboljša točnost osnovnega modela.



Slika 5.5:

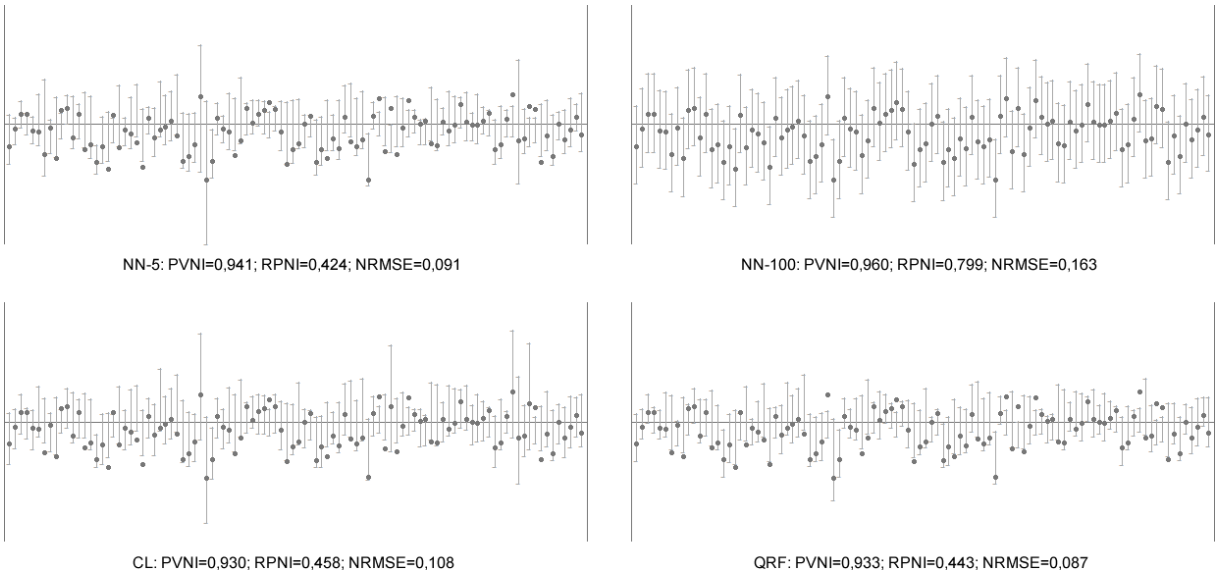
Napovedni intervali na umetnem problemu linearne odsekovne (lomljene) funkcije. Prikazih je 100 napovedi metod BML-A, BML-B, CL in QRF z modelom naključni gozdovi.

Druga primerjava na sliki 5.5 prikazuje napovedi modelov na preprostem problemu odsekovne linearne funkcije. Razlika v ocenah PVNI in RPNI metod BML se opazi na obeh grafih. Napovedni intervali metode BML-A so širši in posledično zajamejo več pravih vrednosti. Vizualno se metodi s pristopom razvrščanja v skupine in kvantilnega regresijskega gozda obnašata zelo podobno.

Me štirimi prikazanimi so najbolj točne napovedi z metodo BML-B. Napovedni intervali s srednjimi vrednostmi modela bagging se nekoliko bolje prilegajo podatkom, kar zaradi manjših

sprememb na grafu ni opazno. Napovedovanje z metodama CL in QRF je na izbrani podatkovni množici manj uspešno.

Primerjavo vseh metod na osnovi lokalnih okolic smo naredili na odsekovno konstantni funkciji. Na sliki 5.6 so prikazani rezultati pri modeliranju z metodo podpornih vektorjev.



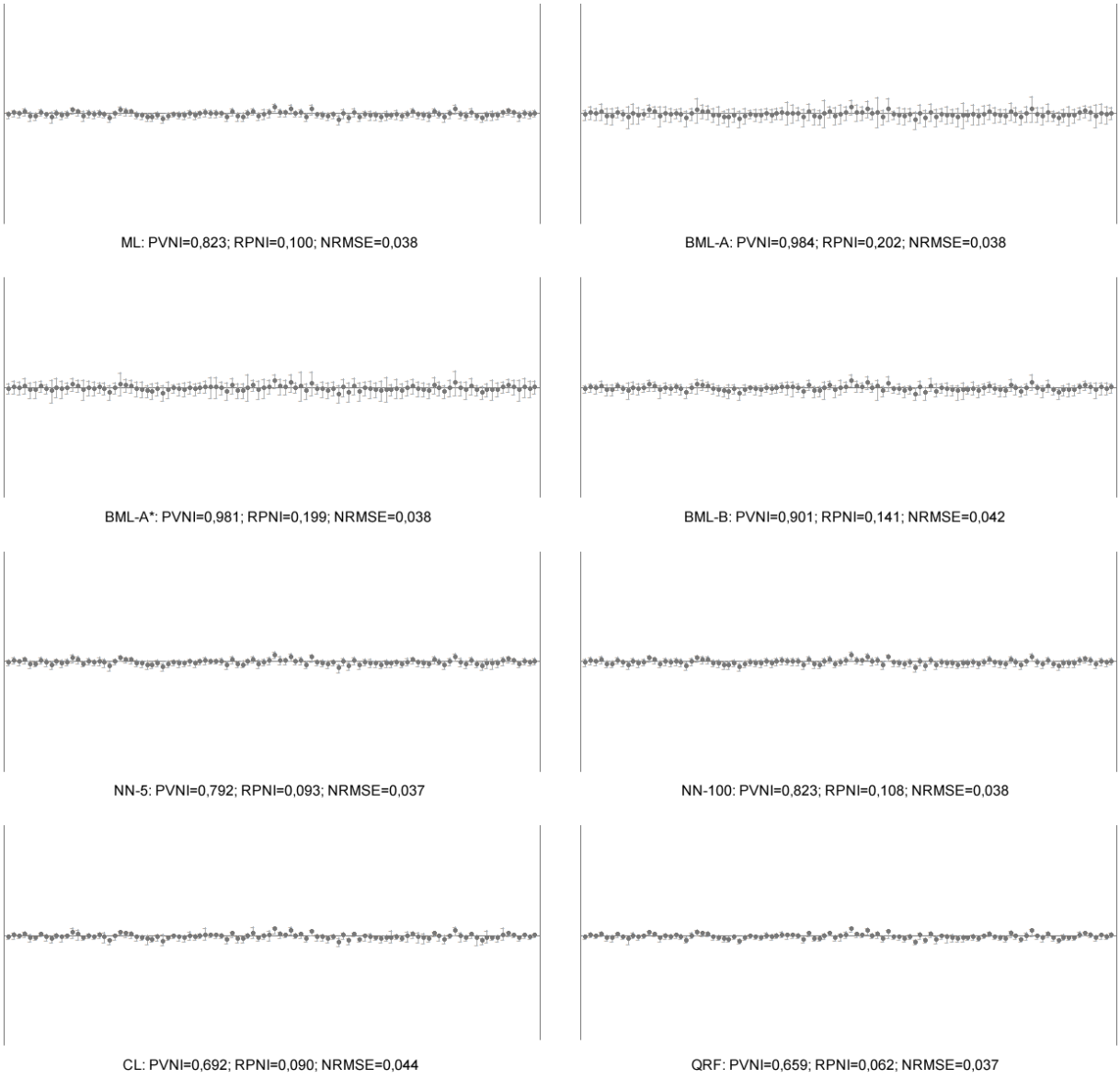
Slika 5.6:

Primerjava intervalnih cenilk na osnovi lokalnih okolic. Prikazih je 100 napovedi metode podpornih vektorjev na problemu odsekoma konstantne funkcije.

Očitno je, da so konstantni napovedni intervali metode NN-100 najmanj informativni. Intervalne ocene ostalih cenilk se bolje prilegajo podatkom. Na grafu metode CL vidimo, da so nekateri napovedni intervali precej širši. Izkaže se, da imajo ti primeri vrednost neodvisne spremenljivke blizu meji med odsekoma stopničaste funkcije, kjer so tudi napake napovedi največje. Pri metodi NN-5, ki upošteva residuale najbližjih sosedov, je ekstremov manj. Na prvi pogled bi ocenili, da metoda QRF poda v povprečju najbolj optimalne intervale, vendar je metoda NN-5 uspešnejša.

Z izjemo konstantnih intervalov napovedovanje z oceno zanesljivosti zmanjša napake napovedi. Najbolj točne napovedne intervale poda kvantilni regresijski gozd. Čeprav metoda QRF pri ocenjevanju zanesljivosti upošteva napoved osnovnega modela, bi iz grafa sklepali, da sta neodvisni.

Vse metode smo preizkusili na umetno generirani podatkovni množici *MV Artificial Domain* [38]. Podatki predstavljajo težji nelinearni problem z 11 odvisnimi numeričnimi in kategoričnimi atributi. Rezultati napovedovanja z najbolj (SVM) in najmanj uspešnim (kNN) regresijskim modelom so predstavljeni na slikah 5.7 in 5.8. Grafi so za lažjo primerjavo prikazani v enakem merilu.

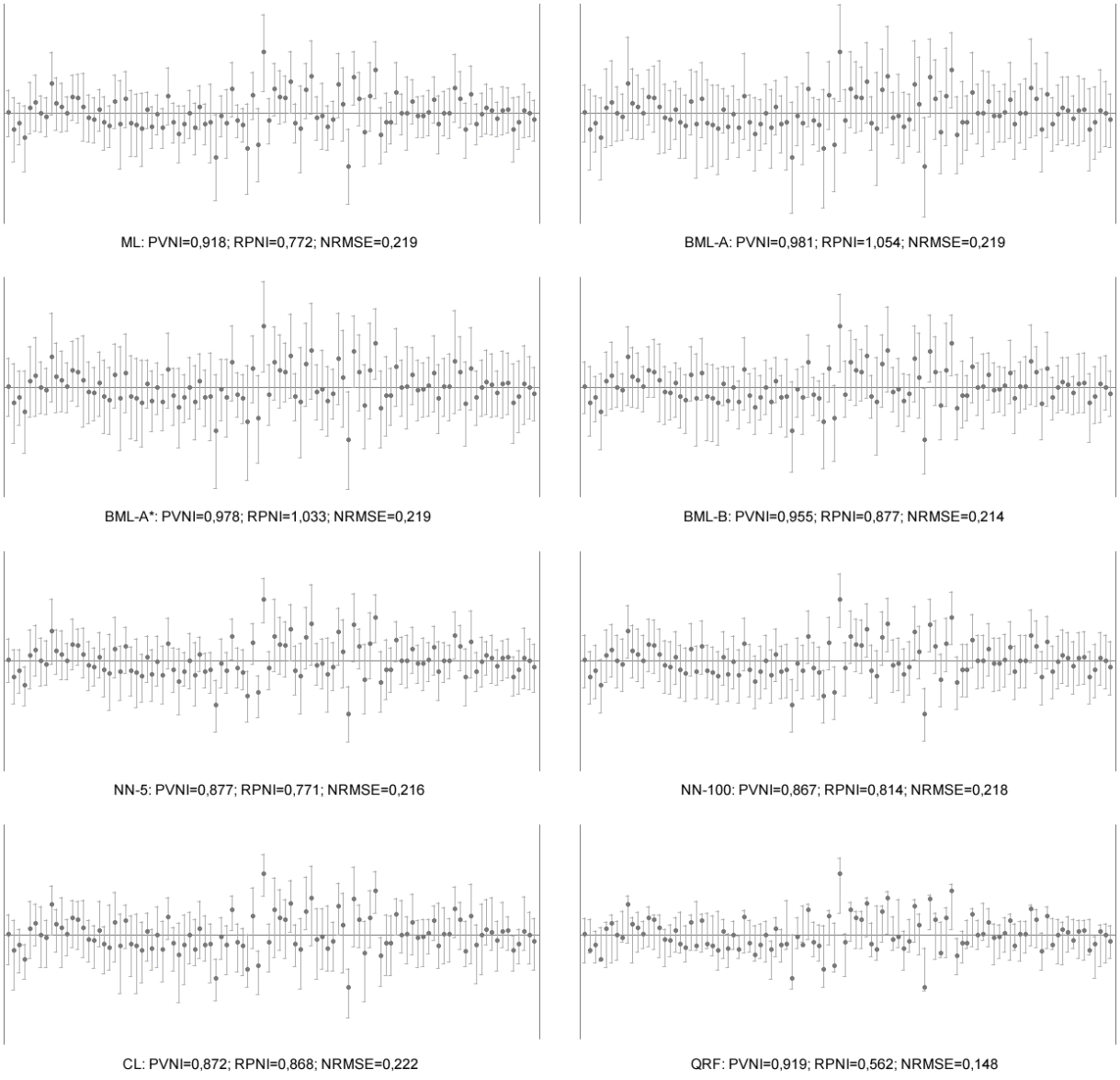


Slika 5.7:

Prikaz napovednih intervalov metode podpornih vektorjev na umetni podatkovni množici MV Artificial domain. Prikazanih je prvih 100 napovedi.

Pri napovedovanju z naprednim modelom metode podpornih vektorjev so vse metode ocenjene z razmeroma nizko vrednostjo RPNI. Kljub temu so intervali obeh različic metode BML-A v povprečju dvakrat širši od ostalih. Ker že osnovni učni algoritem dobro modelira podatke, se večinoma napovedni intervali dobro prilegajo residualom in dosegajo visoko točnost. Izjema je metoda CL, ki na izbranem problemu nekoliko poveča napake napovedi.

Na prikazih napovedi modela k-najbližjih sosedov opazimo, da so intervalne ocene manj optimalne in so razlike med vrednostmi RPNI manjše. Z dobrimi rezultati ponovno izstopa kvantilni regresijski gozd, ki napovedi popravi v smeri dejanskih vrednosti.



Slika 5.8:

Primerjava napovednih intervalov modela k-najbližjih sosedov na umetno generirani podatkovni množici MV Artificial domain.

Poglavje 6 Zaključki

V diplomski nalogi smo raziskali različne pristope k ocenjevanju zanesljivosti napovedovanja na podatkovnih tokovih. Želeli smo poiskati metodo, ki je hitra in v povprečju poda najboljše intervalne ocene na poljubnih regresijskih problemskih množicah z različnimi napovednimi modeli. Rezultati so pokazali, da splošna metoda, ki bi bila po vseh kriterij ocenjena najbolje ne obstaja. Vseeno lahko zaključimo, da so nekatere metode bolj in druge manj primerne za napovedovanje na neskončnih dinamičnih podatkih.

Metode stremljenja tvorijo najbolj pravilne napovedne intervale. To ni nujno dobra lastnost, saj so hkrati najmanj optimalni (najširši). Najbolj točni so napovedni intervali metode, ki uporablja 5% najbližjih sosedov (NN-5). Po oceni skupne statistike sta uspešni tudi metoda razvrščanja v skupine CL in kvantilni regresijski gozd (QRF).

Ko modeliramo podatkovne tokove, je verjetno najbolj stroga omejitev čas izvajanja algoritma. Metode BML s pristopom stremljenja najbolj upočasnijo učni algoritem in so po tem kriteriju očitno najslabša izbira. Pri manjši velikosti drsečega okna se vse ostale metode odrežejo dosti bolje. Med vsemi ima v poprečju najmanjši faktor upočasnitve metoda najbližjih sosedov, ki upošteva residuale celotne učne množice (NN-100). Pri tej se tudi pri največjem oknu izvajanje algoritma upočasni za manj kot četrtino. Pomembna je ugotovitev, da nekatere metode algoritem upočasnijo tudi v fazi napovedovanja. Kadar potrebujemo hitre odločitve, metode s pristopom maksimalnega verjetja in stremljenja ter NN-5 (pri večjem oknu) niso primerne.

Metode na osnovi lokalnih okolic lahko učinkovito uporabimo v kombinaciji s časovno zahtevnejšimi regresijskimi algoritmi. V nekaterih primerih lahko z ocenami zanesljivosti celo zmanjšamo napake napovedi pri enaki velikosti okna. Najučinkovitejša je metoda NN-100, ki je tudi edina, ki jo lahko ob doslednem upoštevanju časovnih omejitev uporabimo z vsemi napovednimi modeli. Metode stremljenja so v tem pogledu praktično neuporabne.

Po drugi strani so konstantni napovedni intervali cenilke NN-100 najmanj informativni. Intervali metod s pristopom maksimalnega verjetja in stremljenja se bolje prilegajo residualom. Ker so simetrični okoli napovedi, so algoritmi z oceno zanesljivosti enako uspešni kot osnovni model. Nasprotno pa lahko intervalne ocene na osnovi lokalnih okolic spremenijo napovedi modela in se bolje prilagodijo dejanskim podatkom. Ali se s tem izboljša točnost napovedi, je odvisno od posameznega problema in/ali regresijskega algoritma. V naših testih se je kot najuspešnejša metoda v tem pogledu izkazala QRF. Razlike med posameznimi pristopi so bolj vidne, kadar model slabše modelira podatke. Pričakovano so takrat napovedni intervali vseh metod širši.

Za napovedovanje zanesljivosti na podatkovnih tokovih so primernejše metode na osnovi lokalnih okolic. Testi in meritve, ki smo jih naredili v sklopu diplomske naloge, so premalo izčrpne, da bi lahko nekritično določili najboljšo metodo. Podatki v podatkovnih tokovih so lahko zelo raznoliki. Od preprostih meritev senzorjev do kompleksnih hitro spreminjajočih podatkov. Tudi v preizkusih smo na različnih množicah podatkov z enakimi metodami dobili različno dobre rezultate.

Za boljšo primerljivost bi bilo potrebno teste ponoviti z različnimi parametri metod (npr. različno število učnih množic modela bagging, izbira drugega modela za napovedovanje variance šuma podatkov…). Na naših testnih množicah se je kot najboljša izbira pokazala metoda na osnovi lokalnih okolic s 5% najbližjih sosedov, kot drugo bi izpostavili metodo s kvantilnim regresijskim gozdom. Gotovo so za delo na podatkovnih tokovih najmanj primerne metode s pristopom stremljenja.

6.1 Nadaljnje delo

Testiranja metod smo se lotili s statičnim učenjem na učnih množicah. Za modeliranje podatkovnih tokov se pogosto uporabijo naprednejše inkrementalne različice algoritmov strojnega učenja, ki so časovno bistveno učinkovitejše. Na enak način je mogoče pohitriti tudi nekatere metode za ocenjevanje zanesljivosti. Metode stremljenja, ki izvajanje učnega algoritma najbolj upočasnijo, lahko implementiramo z uporabo sprotne različice modela bagging in inkrementalnim učenjem notranjih modelov. Poleg manjšega izvajalnega časa lahko takšen pristop izboljša tudi točnost napovedi, saj modeli ohranijo staro znanje.

Pri uporabi metod smo predvideli fiksne velikosti drsečega okna. Adaptivno drseče okno (angl. *adaptive windowing*) lahko prilagodi svojo velikost, pri čemer se zoži ali razširi. V delu [6] je ta pristop uporabljen na podatkih, ki se spreminjajo skozi čas. Če se medprihodni čas podatkov spreminja, lahko, kadar so potrebne hitrejše odločitve, učenje izvajamo na manjši učni množici ali uporabimo hitrejše intervalne cenilke.

Obravnavali smo metode, ki jih lahko uporabimo z vsemi regresijskimi problemi. V [14] je predstavljena novejša ne-parametrična metoda, ki jo v diplomskem delu nismo zajeli. V povezavi z nekaterimi regresijskimi modeli bi lahko uporabili druge specifične metode intervalnega ocenjevanja zanesljivosti. Prav tako so bile razvite napredne metode za uporabo v posebnih aplikacijah [3].

Pri tvorjenju intervalne ocene lahko kombiniramo napovedi različnih metod ocenjevanja zanesljivosti. Čeprav bi s tem algoritem dodatno upočasnili, bi lahko dobili bolj pravilne in optimalne napovedne intervale. Različne pristope lahko združimo tudi tako, da jih uporabimo izmenično. Podatki v podatkovnem toku se lahko dinamično spreminjajo. V določenem obdobju je bolje uporabiti en pristop, v drugem obdobju drugi.

V naših poskusih so bili vsi primeri iz ene testne podatkovne množice vzorčeni iz istega regresijskega problema. V podatkovnih tokovih lahko prihaja do nenadnih sprememb v ciljnem konceptu. Ocene zanesljivosti bi lahko uporabili za prepoznavanje teh sprememb. Za oceno njihove uspešnosti bi bilo potrebno intervalne cenilke ovrednotiti na več realnih in umetnih spremenljivih problemih.

Med obdelavo podatkovnih tokov je lahko omejitev tudi prostorska, zaradi česar nekaterih metod ne moremo uporabiti. V naših poskusih so za izvajanje največ prostora potrebovale metode s pristopom stremljenja, ki pa bi jih lahko z boljšo implementacijo dodatno optimizirali.

Literatura

[1] S. H. A. Ali, K. Fukase in S. Ozawa, "A fast online learning algorithm of radial basis function network with locality sensitive hashing", *Evolving Systems*, vol. 7, 2016, str. 1-14.

[2] G. Arfken, "Taylor's Expansion", *Mathematical Methods for Physicists*, Orlando: Academic Press, 1985, str. 303-313.

[3] J. S. Armstrong, "Prediction Intervals for Time-Series Forecasting", v *Principles of Forecasting: A Handbook for Researchers and Practitioners*, vol. 30, 2001, str. 475-494.

[4] B. Babcock, M. Datar, R. Motwani in L. O'Callaghan, "Sliding Window Computations over Data Streams", *Technical Report*, Stanford University, 2002.

[5] K. Bache and M. Lichman, "UCI machine learning repository", Donald Bren School of Information and Computer Sciences. Dosegljivo: http://archive.ics.uci.edu/ml. [dostopano: 15.5.2016].

[6] A. Bifet in R. Gavaldà, "Learning from Time-Changing Data with Adaptive Windowing", *Proceedings of the 7th SIAM International Conference on Data Mining*, 2007, str. 1-17.

[7] Z. Bosnić, J. Demšar, G. Kešpret, P. P. Rodrigues, J. Gama in I. Kononenko, "Enhancing data stream predictions with reliability estimators and explanation", *Engineering Applications of Artificial Intelligence*, vol. 34, 2014, str. 178-192.

[8] Z. Bosnić, P. P. Rodrigues, I. Kononenko in J. Gama, "Correcting Streaming Predictions of an Electricity Load Forecast System Using a Prediction Reliability Estimate", *Man-Machine Interactions 2*, vol. 103, 2011, str. 343-350.

[9] L. Bruzzone in D. F. Prieto, "An incremental-learning neural network for the classification of remote-sensing image", *Pattern Recognition Letters 20*, vol. 20, 1999, str. 1241-1248.

[10] G. Cauwenberghs in T. Poggio, "Incremental and Decremental Support Vector Machine Learning", *Advances in Neural Information Processing Systems (NIPS 2000)*, vol. 13, 2000, str. 400-415.

[11] A. Dhurandhar in M. Petrik, "Efficient and Accurate Methods for Updating Generalized Linear Models with Multiple Feature Additions", *Journal of Machine Learning Research*, vol. 15, 2014, str. 2607-2627.

[12] B. Efron in R. Tibshirani, "An Introduction to the Bootstrap", Boca Raton: Chapman & Hall/CRC, 1993.

[13] K. Förster, S. Monteleone, A. Calatroni, D. Roggen in G. Törster, "Incremental kNN classifier exploiting correct - error teacher for activity recognition", *Proceedings of the 9th international conference on Machine Learning and Applications (ICMLA)*, 2010, str. 445-450.

[14] J. Frey, "Data-driven nonparametric prediction intervals", *Journal of Statistical Planning and Inference*, vol. 143, 2013, str. 1039–1048.

[15] J. Gama, "Knowledge discovery from data streams", Chapman and Hall/CRC, 2010.

[16] A. Gholipour, M. Javad Hosseini, in H. Beigy, "An adaptive regression tree for non-stationary data streams", *Proceedings of the 28th Annual ACM Symposium on Applied Computing*, 2013, str. 815-817.

[17] J. A. Hartigan in M. A. Wong, "Algorithm AS 136: A K-Means Clustering Algorithm", *Journal of the Royal Statistical Society. Series C (Applied Statistics)*, vol. 28, 1979, str. 100-108.

[18] T. Heskes, "Practical Confidence and Prediction Intervals", *Advances in Neural Information Processing Systems 9*, 1997, str. 176-182.

[19] H. Hu, D. L. Lee in J. Xu, "Fast Nearest Neighbor Search on Road Networks", *Advances in Database Technology - EDBT 2006*, vol. 3896, 2006, str. 186-203.

[20] E. Ikonomovska, "Algorithms for Learning Regression Trees and Ensembles on Evolving Data Streams", doktorska disertacija, Institut Jožef Stefan, Mednarodna podiplomska šola Jožefa Stefana, 2012.

[21] P. Laskov, C. Gehl, S. Kröger in K. R. Müller, "Incremental Support Vector Learning: Analysis, Implementation and Applications", *Journal of Machine Learning Research 7*, 2006, str. 1909–1936.

[22] J. Leskovec, A. Rajaraman, J. D. Ullman, "Mining of massive datasets", Cambridge University Press, 2014, str. 123-153.

[23] Y. Lin in Y. Jeon, "Random forests and adaptive nearest neighbors", *Journal of the American Statistical Association*, 2002, str. 101-474.

[24] N. Meinshausen, "Quantile regression forests", *Journal of Machine Learning Research*, vol. 7, 2006, str. 983-999.

[25] I. J. Myung, "Tutorial on maximum likelihood estimation", *Journal of Mathematical Psychology*, vol. 47, 2003, str. 90-100.

[26] C. H. Nadungodage, Y. Xia, F. Li in J. Ge, "StreamFitter: A Real Time Linear Regression Analysis System for Continuous Data Streams", *Database Systems for Advanced Applications*, vol. 6588, 2011, str. 458-461.

[27] N. C. Oza in Stuart Russell, "Online Bagging and Boosting", *IEEE International Conference on Systems, Man and Cybernetics*, vol. 3, 2005, str. 2340-2345.

[28] D. Pevec, "Ocenjevanje zanesljivosti posameznih napovedi pri nadzorovanem učenju", doktorska disertacija, Univerza v Ljubljani, Fakulteta za računalništvo in informatiko, 2013.

[29] D. Pevec in I. Kononenko, "Input dependent prediction intervals for supervised regression", *Intelligent Data Analysis*, vol. 18, 2014, str. 873-887.

[30] R. Polikar, L. Udpa, S. S. Udpa in V. Honavar, "Learn++: An Incremental Learning Algorithm for Supervised Neural Networks", *IEEE transactions on Systems, Man, and Cybernetics - Part B*, vol. 31, 2001, str. 497-508.

[31] C. E. Rasmussen, R. M. Neal, G. Hinton, D. van Camp, M. Revow, Z. Ghahramani, R. Kustra in R. Tibshirani, "Dvelve - Data for Evaluating Learning in Valid Experiments", University of Toronto, Department of Computer Science. Dosegljivo: http://www.cs.toronto.edu/~delve. [dostopano: 15.5.2016].

[32] P. P. Rodrigues, J. Gama in Z. Bosnić, "Online Reliability Estimates for Individual Predictions in Data Streams", *Workshops Proceedings of the 8th IEEE International Conference on Data Mining*, 2008, str. 36-45.

[33] A. Saffari, C. Leistner, J. Santner, M. Godec in H. Bischof, "On-line Random Forests", *IEEE 12th International Conference on Computer Vision Workshops (ICCV Workshops)*, 2009, str. 1393-1400.

[34] J. C. Schlimmer in D. Fisher, "A case study of incremental concept induction", *Proceedings of the 5th National Conference on Artificial Intelligence*, 1986, str. 496-501.

[35] G. A. F. Seber in C. J. Wild, "Nonlinear Regression", New York: John Wiley & Son, 2003, str. 191-270.

[36] D. L. Shrestha in D. P. Solomatine, "Machine learning approaches for estimation of prediction interval for the model output", *Neural Networks*, vol. 19, 2006, str. 225–235.

[37] A. L. Strehl in M. L. Littman, "Online Linear Regression and Its Application to Model-Based Reinforcement Learning", *Advances in Neural Information Processing Systems 20*, 2007, str. 1417-1424.

[38] L. Torgo, "Regression Data Sets - LIAAD", University of Porto, Laboratory of Artificial Intelligence and Computer Science. Dosegljivo: http://www.dcc.fc.up.pt/~ltorgo/Regression/DataSets.html. [dostopano: 15.5.2016].

[39] P. E. Utgoff, N. C. Berkman, J. A. Clouse in D. Fisher, "Decision tree induction based on efficient tree restructuring", *Machine Learning*, vol. 29, 1997, str. 5-44.

[40] P. Vlachos, "StatLib - Datasets Archive", Carnegie Mellon University, Department of Statistics. Dosegljivo: http://lib.stat.cmu.edu/datasets. [dostopano: 15.05.2016].

[41] A. Zapranis in E. Livanis, "Prediction intervals for neural network models", *Proceedings of the 9th WSEAS International Conference on Computers*, 2005, str. 1-7.